

Streams

Version 3.03

User's Guide:
Core Objects

by Dr. Roger Nokes

Streams

Version 3.03

User's Guide: Core Objects





© Dr Roger Nokes
February 2021

Department of Civil and Natural Resources Engineering
University of Canterbury
Christchurch, NZ
roger.nokes@canterbury.ac.nz

Design and layout: Melody Callahan

1 Introduction 1

2 Streams 3

2.1 Description 3

2.2 Views 3
2.2.1 Application view 3
2.2.2 Object list view 14
2.2.3 Image sequence list view 15
2.2.4 Particle record list view 16
2.2.5 Particle record bundle list view 16
2.2.6 Eulerian field list view 17
2.2.7 Tablet list view 18
2.2.8 Lagrangian field list view 18
2.2.9 Process list view 19
2.2.10 Field list view 19
2.2.11 Manual list view 20
2.2.12 Diagnostic list view 21
2.2.13 Field registry list view 21

3 Image Sequences 23

3.1 Description 23

3.2 Views 24
3.2.1 Details view 24
3.2.2 Image view 26
3.2.3 Process view 35

3.3 Popup menu 36

3.4 Creating image sequences 37
3.4.1 Using standard images 37
3.4.2 Using JAI images 39

4 Particles in 2D 43

4.1 Description 43

4.2 Views 44
4.2.1 Details view 44
4.2.2 Intensity matrix view 44

5 Particle Frames in 2D 47

5.1 Description 47

5.2 Views 47
5.2.1 Details view 47
5.2.2 Particle list view 48
5.2.3 Matches view 48
5.2.4 Segmentation view 49

5.3 Popup menu 50

6 Particle Records in 2D 51

6.1 Description 51

6.2 Views 51
6.2.1 Details View 52
6.2.2 Particle view 53
6.2.3 Process view 66
6.2.4 Match summary view 66
6.2.5 Comparison view 66
6.2.6 Diagnostic view 75
6.2.7 Particle details view 78
6.2.8 Match details view 78
6.2.9 Candidate matches view 79
6.2.10 Particle path view 80

6.3 Popup menu 81

7 Particles in 3D 85

7.1 Description 85

7.2 Views 85
7.2.1 Details view 86
7.2.2 Path view 86

8 Particle Frames in 3D 87

8.1 Description 87

8.2 Views 87
8.2.1 Details view 87
8.2.2 Particle list view 88

8.2.3 Matches view 88
8.2.4 Segmentation view 89

8.3 Popup menu 90

9 Particle Records in 3D 91

9.1 Description 91

9.2 Views 91
9.2.1 Details View 92
9.2.2 Particle view 93
9.2.3 Process view 104
9.2.4 Match summary view 104
9.2.5 Comparison view 104
9.2.6 Diagnostic view 113
9.2.7 Particle list view 115
9.2.8 Match details view 116

9.3 Popup menu 116

10 Particle 2D Record Bundles 121

10.1 Description 121

10.2 Views 122
10.2.1 Details View 122
10.2.2 Process view 123

10.3 Popup menu 123

10.4 Perspectives 124
10.4.1 Details View 124
10.4.2 Popup menu 125

10.5 Perspective cameras 125
10.5.1 Details View 126
10.5.2 Calibration view 128
10.5.3 Popup menu 131

10.6 Model cameras 131
10.6.1 Details View 132
10.6.2 Popup menu 134

10.7 Creating particle 2D record bundles 134

10.8 Generate particle 2D record bundle process 135
10.8.1 Details view 136
10.8.2 Popup menu 137

11 Intensity Field Time Series 139

11.1 Description 139

11.2 Views 139
11.2.1 Details view 139
11.2.2 Calculator view 140
11.2.3 Process view 141

11.3 Popup menu 141

11.4 Calculators 142
11.4.1 Manager View 142
11.4.2 Custom calculators 144
11.4.3 Popup menu 144

12 Velocity 2D Field Time Series 147

12.1 Description 147

12.2 Views 147
12.2.1 Details view 147
12.2.2 Calculator view 148
12.2.3 Process view 149

12.3 Popup menu 149

12.4 Calculators 150
12.4.1 Manager View 152
12.4.2 Custom calculators 152
12.4.3 Popup menu 152

13 Velocity 3D Field Time Series 153

13.1 Description 153

13.2 Views 153
13.2.1 Details view 153
13.2.2 Calculator view 154
13.2.3 Process view 155

13.3 Popup menu 155

13.4 Calculators 156
13.4.1 Manager View 158
13.4.2 Custom calculators 160
13.4.3 Popup menu 160

14 Velocity-Intensity Field Time Series 161

14.1 Description 161

14.2 Views 161
14.2.1 Details view 162
14.2.2 Calculator view 162
14.2.3 Process view 163

14.3 Popup menu 163

14.4 Calculators 164
14.4.1 Manager View 166
14.4.2 Custom calculators 166
14.4.3 Popup menu 166

14.5 Creating velocity-intensity field time series 167

15 Density Field Time Series 169

15.1 Description 169

15.2 Views 169
15.2.1 Details view 169
15.2.2 Calculator view 170
15.2.3 Process view 170

15.3 Popup menu 171

15.4 Calculators 171
15.4.1 Manager View 172
15.4.2 Popup menu 172

16 Shape Field Time Series 173

16.1 Description 173

16.2 Views 173
16.2.1 Details view 173
16.2.2 Calculator view 174
16.2.3 Process view 174

16.3 Popup menu 175

16.4 Calculators 175
16.4.1 Manager View 176
16.4.2 Popup menu 176

17 Displacement 2D Field Time Series 177

17.1 Description 177

17.2 Views 177
17.2.1 Details view 178
17.2.2 Calculator view 178
17.2.3 Process view 179

17.3 Popup menu 179

17.4 Calculators 180
17.4.1 Manager View 181
17.4.2 Custom calculators 181
17.4.3 Popup menu 181

18 Displacement 3D Field Time Series 183

18.1 Description 183

18.2 Views 183
18.2.1 Details view 184
18.2.2 Calculator view 184
18.2.3 Process view 185

18.3 Popup menu 185

18.4 Calculators 186
18.4.1 Manager View 187
18.4.2 Custom calculators 187
18.4.3 Popup menu 187

19 Displacement 3D Surface Field Time Series 189

19.1 Description 189

19.2 Views 189
19.2.1 Details view 190
19.2.2 Calculator view 190
19.2.3 Process view 191

19.3 Popup menu 191

19.4 Calculators 192
19.4.1 Manager View 193
19.4.2 Custom calculators 193
19.4.3 Popup menu 193

20 Lagrangian 2D Path Fields 195

20.1 Description 195

20.2 Views 195
20.2.1 Calculator view 196
20.2.2 Details view 197
20.2.3 Particles view 197
20.2.4 Path view 198
20.2.5 Path list view 205
20.2.6 Process view 206
20.2.7 Selected path list view 206

20.3 Popup menu 207

20.4 Calculators 207
20.4.1 Manager View 209
20.4.2 Custom calculators 210
20.4.3 Two path calculators 210
20.4.4 Popup menu 211

21 Lagrangian 3D Path Fields 213

21.1 Description 213

21.2 Views 213
21.2.1 Calculator view 214
21.2.2 Details view 215
21.2.3 Particles view 215
21.2.4 Path view 215
21.2.5 Path list view 224
21.2.6 Process view 225
21.2.7 Selected path list view 225

21.3 Popup menu 225

21.4 Calculators 226
21.4.1 Manager View 228

21.4.2 Custom calculators 229
21.4.3 Two path calculators 230
21.4.3 Popup menu 231

22 Free Processes 233

22.1 Description 233
22.1.1 Popup menu 233

22.2 Image processes 233
22.2.1 Overlay images process 234
22.2.2 Average image sequences process 235
22.2.3 Image sequence auto generation process 236
22.2.4 Create constant colour image process 239

22.3 Particle record processes - models 240

22.4 Particle record processes - other 246
22.4.1 Overlay particle 2D records process 246
22.4.2 Create camera 3D calibration image process 247
22.4.3 Create 3D calibration target process 249

22.5 Lagrangian path field processes 251
22.5.1 Overlay Lagrangian path fields 2D process 251
22.5.2 Transform from one path field 2D process 252
22.5.3 Transform from two path fields 2D process 254

22.6 Eulerian field processes – combining fields 256
22.6.1 Concatenate Eulerian field process 256
22.6.2 Overlay Eulerian field process 257
22.6.3 Subtract Eulerian field process 258

22.7 Eulerian field processes – other 259
22.7.1 Create model velocity 3D field process 259

22.8 Miscellaneous processes 261
22.8.1 Batch process 261

23 Image Filter Pipelines 263

23.1 Description 263

23.2 Views 263
23.2.1 Details view 264
23.2.2 Filter view 264

23.3 Popup menu 265

23.4 Filters 265
23.4.1 Popup menu 266
23.4.2 Details View 266
23.4.3 Absorption colour filter 267
23.4.4 Absorption image filter 267
23.4.5 Amplify filter 268
23.4.6 Barrel pincushion distortion filter 269
23.4.7 Bayer colour filter 270
23.4.8 Convert type filter 270
23.4.9 Convolution filter 271
23.4.10 Discrimination 1 filter 272
23.4.11 Discrimination 2 filter 273
23.4.12 Eliminate objects filter 274
23.4.13 Extract filter 275
23.4.14 FFT filter 276
23.4.15 Identity filter 276
23.4.16 Invert filter 277
23.4.17 Mathematical formula filter 277
23.4.18 Overwrite regions filter 278
23.4.19 Parallax factor filter 279
23.4.20 Polynomial field calibration filter 279
23.4.21 Remove filter 282
23.4.22 Remove background filter 282
23.4.23 Scale by maximum in regions filter 283
23.4.24 Spatial average filter 284
23.4.25 Subtract colour filter 285
23.4.26 Subtract image filter 285
23.4.27 Subtract image sequence filter 286
23.4.28 Threshold filter 286
23.4.29 White balance image filter 287

24 Particle Filter 2D Pipelines 289

24.1 Description 289

24.2 Views 289
24.2.1 Details view 289
24.2.2 Filter view 290

24.3 Popup menu 290

24.4 Filters 291
24.4.1 Popup menu 291
24.4.2 Details View 291
24.4.3 Delete tracked regions 292

25 Particle 2D Identification Processes 295

25.1 Description 295

25.2 Simple threshold mono PID 295
25.2.1 Algorithm data 296

25.3 Simple threshold colour PID 296
25.3.1 Algorithm data 297

25.4 Dual relative threshold PID 297
25.4.1 Algorithm data 298

25.5 Dual threshold Gaussian PID 299
25.5.1 Algorithm data 300

25.6 Common parameters 301

25.7 Popup menu 302

26 PTV 2D Analysis Pipelines 303

26.1 Description 303

26.2 Views 303
26.2.1 Details view 303
26.2.2 Analysis view 304

26.3 Popup menu 304

26.4 PTV 2D analysis 305
26.4.1 Details view 305
26.4.2 Popup menu 308

26.5 Costings in 2D 309
26.5.1 Details View 309
26.5.2 Adjacency costing 310
26.5.3 Centre of mass costing 310
26.5.4 Colour costing 311
26.5.5 Correlation costing 311
26.5.6 Distance costing 312

26.5.7 Intensity costing 313
26.5.8 Pseudo-correlation costing 313
26.5.9 Shape costing 314
26.5.10 Direction velocity costing 315
26.5.11 Least squares velocity costing 315
26.5.12 Local velocity costing 316
26.5.13 Magnitude velocity costing 318
26.5.14 Path length costing 318
26.5.15 Polynomial velocity costing 318
26.5.16 Recent acceleration costing 319
26.5.17 Recent velocity costing 319
26.5.18 Space average acceleration costing 320
26.5.19 Time average velocity costing 320

27 PTV 3D Analysis Pipelines 323

27.1 Description 323

27.2 Views 323
27.2.1 Details view 323
27.2.2 Analysis view 324

27.3 Popup menu 324

27.4 PTV 3D analysis 325
27.4.1 Details view 325
27.4.2 Popup menu 328

27.5 Costings in 3D 329
27.5.1 Details View 329
27.5.2 Adjacency costing 330
27.5.3 Centre of mass costing 331
27.5.4 Colour costing 331
27.5.5 Distance costing 332
27.5.6 Intensity costing 332
27.5.7 Pseudo-correlation costing 332
27.5.8 Direction velocity costing 333
27.5.9 Least squares velocity costing 334
27.5.10 Local velocity costing 335
27.5.11 Magnitude velocity costing 336
27.5.12 Path length costing 337
27.5.13 Polynomial velocity costing 337
27.5.14 Recent acceleration costing 337

27.5.15 Recent velocity costing 338
27.5.16 Space average acceleration costing 338
27.5.17 Time average velocity costing 339

28 Bundle PTV Analysis Pipelines 341

28.1 Description 341

28.2 Views 341
28.2.1 Details view 341
28.2.2 Analysis view 343

28.3 Popup menu 343

28.4 Bundle matching analysis 344
28.4.1 Details view 344
28.4.2 Popup menu 345

28.5 Costings 345
28.5.1 Details View 346
28.5.2 Colour costing 346
28.5.3 Distance costing 347
28.5.4 Intensity costing 347
28.5.5 Mapping error costing 347
28.5.6 Regions costing 348
28.5.7 Size costing 348

29 Predict and Verify Process 349

29.1 Description 349

29.2 Views 350
29.2.1 Details view 350

29.3 Popup menu 351

30 Create Intensity Field Process 353

30.1 Description 353

30.2 Views 354
30.2.1 Details view 354

30.3 Popup menu 356

31 Create Velocity 2D Field Process 357

31.1 Description 357

31.2 Views 358
31.2.1 Details view 358

31.3 Popup menu 362

32 Create Velocity 3D Field Process 363

32.1 Description 363

32.2 Views 363
32.2.1 Details view 364

32.3 Popup menu 368

33 Create Density Field Process 369

33.1 Description 369

33.2 Views 369
33.2.1 Details view 369

33.3 Popup menu 371

34 Create Shape Field Process 373

34.1 Description 373

34.2 Views 373
34.2.1 Details view 374

34.3 Popup menu 377

35 Create Displacement 2D Field Process 379

35.1 Description 379

35.2 Views 380
35.2.1 Details view 380

35.3 Popup menu 384

36 Create Displacement 3D Field Process 385

36.1 Description 385

36.2 Views 385

36.2.1 Details view 385
36.3 Popup menu 389

37 Create Displacement 3D Surface Field Process. 391

37.1 Description 391
37.2 Views 391
37.2.1 Details view 391
37.3 Popup menu 394

38 Create Lagrangian 2D Path Field Process 397

38.1 Description 397
38.2 Views 397
38.2.1 Details view 397
38.3 Popup menu 399

39 Create Lagrangian 3D Path Field Process 401

39.1 Description 401
39.2 Views 401
39.2.1 Details view 401
39.3 Popup menu 403

40 Transform Image Sequence Processes 405

40.1 Description 405
40.2 Trim transform 405
40.2.1 Details View 405
40.2.2 Popup menu 406
40.3 Scale transform 407
40.3.1 Details View 407
40.3.2 Popup menu 407
40.4 Average transform 408
40.4.1 Details View 408
40.4.2 Popup menu 409
40.5 Rotate transform 409

40.5.1 Details View 409
40.5.2 Popup menu 410

40.6 Subtract transform 410
40.6.1 Details View 410
40.6.2 Popup menu 411

40.7 Long exposure transform 411
40.7.1 Details View 412
40.7.2 Popup menu 412

41 Transform Particle 2D Record Processes 413

41.1 Description 413

41.2 Views 413
41.2.1 Details View 413
41.2.2 Transform view 414

41.3 Popup menu 415

41.4 Transforms 415
41.4.1 Views 415
41.4.2 Popup menus 415
41.4.3 Adjust camera 415
41.4.4 Barrel pincushion distortion 416
41.4.5 Constant velocity 417
41.4.6 Swap x 417
41.4.7 Swap y 417
41.4.8 Remove camera movement 417
41.4.9 Map coordinates 418
41.4.10 Transform coordinates 419

42 Transform Particle 3D Record Processes 421

42.1 Description 421

42.2 Views 421
42.2.1 Details View 421
42.2.2 Transform view 422

42.3 Popup menu 423

42.4 Transforms 423
42.4.1 Views 423

42.4.2 Popup menus 423
42.4.3 Linear transform 423
42.4.4 Rotate transform 424
42.4.5 Swap x transform 424
42.4.6 Swap y transform 424
42.4.7 Swap z transform 425
42.4.8 Constant velocity transform 425
42.4.9 Remove camera movement transform 425

43 Transform Eulerian Field Processes 427

43.1 Description 427

43.2 Views 427
43.2.1 Details View 427
43.2.2 Transform view 428

43.3 Popup menu 428

43.4 Transforms 429
43.4.1 Views 429
43.4.2 Popup menus 429
43.4.3 Linear transform 429
43.4.4 Swap x transform 430
43.4.5 Swap y transform 430
43.4.6 Swap z transform 431
43.4.7 Subset transform 431
43.4.8 Average fields in time transform 432
43.4.9 Constant velocity transform 432
43.4.10 File displacements transform 433

44 Transform Lagrangian Path Fields Processes 435

44.1 Description 435

44.2 Views 435
44.2.1 Details View 435
44.2.2 Transform view 436

44.3 Popup menu 436

44.4 Transforms 437
44.4.1 Views 437
44.4.2 Popup menus 437

44.4.3 Swap x transform 437
44.4.4 Swap y transform 437
44.4.5 Transform coordinates 438

45 Join Lagrangian Paths Process 439

45.1 Description 439

45.2 Views 440
45.2.1 Details View 440

45.3 Popup menu 442

46 Save Entity Processes 443

46.1 Description 443

46.2 Views 443
46.2.1 Details View 444

46.3 Popup menu 444

47 Regions in 2D 445

47.1 Description 445

47.2 Views 447
47.2.1 Point Details View 448
47.2.2 Line Details View 448
47.2.3 Rectangle Details View 448
47.2.4 Circle Details View 448
47.2.5 Polygon Details View 448
47.2.6 Freehand Details View 449

47.3 Popup menu 449

47.4 Region 2D manager view 449

47.5 Loading regions from text file 450

48 Regions in 3D 453

48.1 Description 453

48.2 Views 454
48.2.1 Point Details View 454
48.2.2 Line Details View 454

48.2.3 Sphere Details View 454
48.2.4 Cylinder Details View 454
48.2.5 Cone Details View 455
48.2.6 Rectangle Details View 455
48.2.7 General Rectangle Details View 455
48.3 Popup menu 456
48.4 Region 3D manager view 456
48.5 Loading regions from text file 457

49 Formulae 461

49.1 Description 461
49.2 General mathematical formulae 463
49.3 Field calculator formulae 463
49.4 Tablet formulae 465

50 Manuals 467

50.1 Description 467
50.2 Views 467
50.2.1 Details View 467
50.2.2 Reader View 467
50.3 Popup menu 469

1 Introduction

Streams is an image processing and particle tracking velocimetry (PTV) system created at the University of Canterbury. The system has been designed specifically to provide a user interface that is both friendly and powerful so that relative novices can rapidly gain familiarity with its capabilities. Both postgraduate and undergraduate research students have used the system effectively within our laboratory.

Four documents describing *Streams* are available. The first, the ***Streams 3.03* – System Theory and Design** manual, provides a general introduction to image processing and particle tracking and describes in detail the theory underpinning the *Streams* implementations of these tools. The second, third and fourth manuals are User Guides. The first of these, this manual, covers the core objects of the *Streams* system. The second, ***Streams 3.03* – User's Guide: Field Objects** covers the field objects, and the third, ***Streams 3.03* – User's Guide: Tablets** covers tablet objects. In these manuals the details of the user interaction mechanisms are provided.

Streams is designed according to the object model of software development. Both the underlying code and the user interface are consistent with this model. While the code is of little concern to the user, the structure of the interface is of interest. This manual covers all of the core objects, not including field and tablet objects, within the *Streams* system and describes the interface provided to the user to interact with these objects. These objects include all entities such as image sequences, particle records, Eulerian fields etc., as well as all processes, free and bound.

Specifically, this manual

- provides the user with detailed information about the various core objects present in *Streams*,
- explains the available object views, the information they display and the menu options each provides,
- introduces the popup menus and the commands they provide, and
- explains the dialog boxes and the parameters they require.

The manual is broken into three parts. Part 1 covers the entities available to the user, part 2 discusses the free and bound processes supported by *Streams*, and part 3 covers miscellaneous support objects. This manual is

not intended to be read sequentially from start to finish. It is primarily a reference manual and interrelated chapters are likely to be read together.

2 Streams

2.1 Description

A *Streams* object is the top-level object that is created when the *Streams* application runs. This object provides the user with the top level of user interaction, including the main menu bar (see application view below).

All entities are managed by the *Streams* object.

2.2 Views

The *Streams* object provides twelve views. These are:

- application view
- object list view
- image sequence list view
- particle record list view
- particle record bundle list view
- Eulerian field list view
- Lagrangian field list view
- tablet list view
- process list view
- field list view
- manual list view
- diagnostic list view
- field registry list view

2.2.1 Application view

Description

The application view is the parent window for the application. It includes the main menu bar and two toolbars located at the top and bottom of the window. The upper toolbar provides a number of buttons that provide shortcuts to common actions contained in the main menus. Each of the buttons has an identifying icon and a tooltip when the cursor is placed over the button. The buttons are, in order from the left,

- open image sequence
- open particle 2D record
- open particle 3D record
- open particle 2D record bundle
- open velocity 2D field time series
- open velocity 3D field time series
- open intensity field time series
- open displacement 2D field time series
- open displacement 3D field time series
- open displacement 3D surface field time series
- open Lagrangian 2D path field
- open Lagrangian 3D path field
- open tablet
- open free process
- open field
- minimise all field views
- tile field views automatically
- tile field views vertically
- tile field views horizontally
- tile field views with custom settings
- stack field views
- shrink all field views
- expand all field views
- cycle forwards through all field views
- cycle backwards through all field views
- open manual

The lower toolbar provides information about the available memory within the system. This is displayed as both a numerical value (in Mb) and as a sliding scale showing the percentage of the total memory that the numerical value represents. If the user clicks or double clicks on the slider the garbage collector will run and attempt to reclaim any unused memory. The total available memory was specified by the user when the application was launched.

When this view is closed the application exits. Any unsaved objects will prompt the user to save them before closing. Note: If the user is running *Streams* on a Mac an operating system level Menu with the application name is provided in the menubar. This menu offers an option to Quit

Streams. This will exit the application, but it will not prompt the user to save unsaved objects. Thus it is advised not to use this menu.

Access

The application view is opened when the application starts.

Menus

Streams supports keyboard short cuts for most menu items in the main menubar. The short cut is given next to the menu item in the list below. The short cuts use one or more of the SHIFT, ALT/ OPTION and CONTROL keys. These are given the symbols ⇧, ⌥, and ^ respectively.

Streams3.03 Menu (Apple Mac only)

- *Quit Streams3.03* – this is equivalent to choosing Quit from the Application menu except the user will not be prompted to save any unsaved objects before exiting. Note that the keyboard shortcut does not work for this menu option.

Application Menu

- *Reclaim memory* – attempts to retrieve unused memory in the system
- *Properties…* – allows the user to access the application properties. These include default directories and memory management parameters (see the *System Theory and Design* Manual). It also provides the user the option to change the "look and feel" of the interface presented to the user. A look and feel defines how the various windows, controls etc. are displayed to the user. The default look and feel will be that of the operating system under which the application is running. On a Windows machine the user interface will look like any other Windows application. On a Macintosh the interface will conform to Apple standards. The user will generally have the option of selecting two other looks and feels. One is called Metal and is a generic Java interface design. The second is Motif, which is the standard look and feel for some X Windows based UNIX systems. The user is free to switch between these different skins at any time. It is worth noting that the *Streams* application view will always retain the native title bar look and feel no matter what skin has been selected, and this is also true of most dialog boxes.
- *Auto close process monitor dialog* – when a process executes in the *Streams* system a process monitor dialog box is displayed that provides the user with information about the actions being taken and the time taken to execute these actions. When the process completes the dialog box closes. This menu option allows the user to specify whether these monitor dialogs should remain open on completion of the process.
- *Quit ^Q* –allows the user to exit the application. The user will be prompted to save unsaved data before exiting.

View Menu

- *Object list view* ⌥⇧*O* – opens an object list view. This view displays all of the objects currently opened in *Streams*. When the system is started this is the only view displayed other than the application view.
- *Image sequence list view* ⌥⇧*I* – opens an image sequence list view. This view displays all of the image sequences (see chapter 3) currently opened in *Streams*.
- *Particle record list view* ⌥⇧*R* – opens a particle record list view. This view displays all of the particle records (see chapters 6 and 9) currently opened in *Streams*.
- *Particle record bundle list view* ⌥⇧*B* – opens a particle record bundle list view. This view displays all of the particle records (see chapter 10) currently opened in *Streams*.
- *Eulerian field list view* ⌥⇧*E* – opens a Eulerian field list view. This view displays all of the Eulerian fields (see chapters 11-20) currently opened in *Streams*. This includes all intensity, velocity, velocity-intensity, density, shape and displacement field time series.
- *Lagrangian field list view* ⌥⇧*L* – opens a Lagrangian field list view. This view displays all of the Lagrangian 2D and 3D path fields (see chapters 20 and 21) currently opened in *Streams*.
- *Tablet list view* ⌥⇧*T* – opens a Tablet list view. This view displays all of the tablets (see *User's Guide: Field Objects* Manual) currently opened in *Streams*.
- *Process list view* ⌥⇧*P* – opens a process list view. This view displays all of the free processes (see chapter 22) currently opened in *Streams*.
- *Field list view* ⌥⇧*F* – opens a field list view. This view displays all of the field objects (see *User's Guide: Field Objects* Manual) currently open in *Streams*.
- *Manual list view* ⌥⇧*M* – opens a manual list view. This view displays all of the field objects (see *User's Guide: Field Objects* Manual) currently open in *Streams*.
- *Diagnostic list view* ⌥⇧*M*D– opens a diagnostic list view. This view displays all of the currently recorded diagnostics.
- *Field registry list view* ⌥⇧*Y* – opens a field registry list view. This view displays all of the field objects (see *User's Guide: Field Objects* Manual) that have been registered within *Streams*.
- *Hide/show toolbars* – hides or shows the toolbars at the top and bottom of the view.
- *Change background colour…* – allows the user to select a different background colour for the application window.

Image Menu

- *Create image sequence…* ^*M* – allows the user to create a new image sequence from a sequence of image files.

- *Create image sequence from JAI images…* ^J – allows the user to create a new image sequence from a sequence of .jai image files. During the creation process each .jai image is loaded and stored in one of the standard image formats as selected by the user.
- *Open image sequence…*– allows the user to open one or more image sequence objects from files with extension *.seq*. The loaded object(s) will be listed in the object and image sequence list views.

Particle Menu

- *Create particle 2D record from text file…* – allows the user to create a particle 2D record from data supplied in a space delimited text file (.txt or .prn). The text file has the following structure: the time step, x origin, y origin, maximum x coordinate and maximum y coordinate appear on the first line of the file (separated by spaces). The second line specifies the number of frames. Then for each frame the number of particles in the frame is specified on a separate line and this is followed by a line containing the data for each particle – its x and y coordinates, its radius and the red, green and blue intensities. No other particle information can be included.
- *Create particle 3D record from text file…* – allows the user to create a particle 3D record from data supplied in a space delimited text file (.txt or .prn). The text file has the same structure as for the 2D particle record except the first line now has the time step, x origin, y origin, z origin, maximum x coordinate, maximum y coordinate and maximum z coordinate appear and for each particle the x, y and z coordinates are given.
- *Create particle 2D record bundle…* ^B – allows the user to create a particle 2D record bundle from a number of particle 2D records and perspective cameras. See section 10.6 for details.
- *Open particle 2D record…* – allows the user to open one or more particle 2D record objects from files with extension *.par or .par2*. The loaded object(s) will be listed in the object and particle record list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open particle 3D record…* – allows the user to open one or more particle 3D record objects from files with extension *.par3*. The loaded object(s) will be listed in the object and particle record list views. The system will check the available memory before opening the file (see above).
- *Open particle 2D record bundle…* – allows the user to open one or more particle 2D record bundle objects from files with extension *.par2bun*. The loaded object(s) will be listed in the object and particle record bundle list views. The system will check the available memory before opening the file (see above).

Eulerian Menu

- *Create velocity-intensity time series …* ^*I* – This menu option opens a dialog box from which the user can select a velocity field time series and intensity field time series from which to create a velocity-intensity field time series. The currently opened intensity and velocity fields are listed and only two compatible fields can be combined. See chapter 14.
- *Open velocity 2D time series …* – allows the user to open one or more velocity 2D field time series objects from files with extension *.vel or .vel2d*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open velocity 3D time series …* – allows the user to open one or more velocity 3D field time series objects from files with extension *.vel3d*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open displacement 2D time series …* – allows the user to open one or more displacement 2D field time series objects from files with extension *.dis or dis2d*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open displacement 3D time series …* – allows the user to open one or more displacement 3D field time series objects from files with extension *dis3d*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open displacement 3D surface time series …* – allows the user to open one or more displacement 3D surface field time series objects from files with extension *dis3dsur*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open intensity time series …* – allows the user to open one or more intensity field time series object(s) from files with extension *.int*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is

sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.

- *Open velocity-intensity time series …* – allows the user to open one or more velocity-intensity field time series objects from files with extension *.vin*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open density time series …* – allows the user to open one or more density field time series objects from files with extension *.den*. The loaded object will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open shape time series …* – allows the user to open one or more shape field time series objects from files with extension *.shp*. The loaded object(s) will be listed in the object and Eulerian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.

Lagrangian Menu

- *Open Lagrangian path field 2D …* – allows the user to open one or more Lagrangian path field objects from a disk file with extension *.lgr or lgr2d*. The opened object(s) will be listed in the object and Lagrangian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Open Lagrangian path field 3D …* – allows the user to open one or more Lagrangian path field objects from a disk file with extension *.lgr3d*. The opened object(s) will be listed in the object and Lagrangian field list views. The system will attempt to estimate whether the available memory is sufficient for the files to be loaded. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.

Tablet Menu

- *Create tablet …* ^*T* – creates a new, blank, tablet and lists it in the object and tablet list views. A dialog box is displayed that enables the user to specify a name for the tablet.
- *Open tablet …* – allows the user to open one or more tablets from disk files with extension *.tab*. The opened object(s) will be listed in the object and tablet list views.
- *Open all tablet analysis views* – this opens the analysis view of all tablets currently open in *Streams*. It is not uncommon for a user to need to work with multiple tablets at the same time and this option, along with those below, is designed to ease the interaction with these tablets.
- *Horizontally tile all tablet analysis views* – this option finds all of the tablet analysis views currently open and tiles them horizontally across the screen to the right of the object list view.
- *Minimise all tablet analysis views* – this option identifies all tablet analysis views currently open, and minimises them.
- *Close all tablet analysis views* – closes all currently open tablet analysis views.

Process Menu

- *Open process …* – allows the user to open one or more free process objects (see chapter 22) from files with extension *.pcf*. The opened object(s) will be listed in the object and process list views.
- *Image processes…* - allows the user to create a free process associated with images. These are described in chapter 22.
- *Particle record processes…* - allows the user to create a free process that combines particle records or generates a particle record from one of the theoretical models. There are 7 of these models in total and these are discussed in chapter 22. A process that can generate calibration images for calibrating cameras associated with particle 2D record bundles is also accessed through this menu and can be found described in chapter 22. The created process object will be listed in the object and process list views.
- *Eulerian field processes…* - allows the user to create a free process that combines Eulerian field objects or creates a velocity 3D field time series based on a theoretical model. For each of the 4 types of 2D Eulerian fields the user can create a process that overlays, concatenates or subtracts two or more of these fields. These processes are discussed in detail in chapter 22. The created process object will be listed in the object and process list views.
- *Miscellaneous processes…* - allows the user to create one of the miscellaneous free processes described in chapter 22. The created process object will be listed in the object and process

Field Menu

- *Open field …* ^⇧F – allows the user to open one or more field objects (see *User's Guide: Field Objects* Manual) from disk files. The opened objects will be listed in the object and field list views (and registered with the field registry). The system will attempt to estimate whether the available memory is sufficient for the specified field files to be opened. If not the user will be warned. However the user retains the option to override the warning and attempt to open the files anyway. If they choose to do this they should ensure that all currently opened datasets are saved first.
- *Construct field from text file* – this menu has four submenus that enable the user to create four different types of fields from text files. These text files a space delimited. The four field types are scalar field 1D, scalar field 2D, vector 2D field 1D, and vector 2D field. In each case a dialog box is displayed that prompts the user for the text files containing the field data. For 1D fields a file containing the x values of the grid is required while for 2D fields two separate files, one containing the x grid values and the other the y grid values are required. For scalar fields a single text file containing the scalar field values must be provided, while for vector fields two separate field files, one for the x component and the other the y component are required. In the case of a 2D field these field values are ordered by x (i.e. the data are read in by rows – y is constant). The resulting fields are displayed in the object list.
- *Minimise all field views* ^⇧M – it is common for the user to have many field views open simultaneously. This menu option minimizes all of these views to icons that appear along the bottom of the application view window. A shortcut is available in the view's top menubar.
- *Tile all field views*– all field views currently open (whether minimized or not) will be displayed in a tiled fashion (i.e. in rows and columns). All views will be visible. The user has 4 choices for the tile arrangement *Tile field views automatically* (^⇧A) will lay out the views in a rectangular grid that is as close to square as possible. *Tile field views vertically* (^⇧V) will tile the views in one column. *Tile field views horizontally* (^⇧H) will tile the views into a single row. *Tile field views with custom settings* (^⇧S) will tile the views into a rectangular grid with the number of rows specified by the user. All of these actions have shortcuts available in the view's top menubar.
- *Stack all field views*^⇧F – all field views currently open (whether minimized or not) will be displayed as a stack. Only the last view in the stack will be completely visible but the title bars of all windows will be accessible. A shortcut is available in the view's top menubar.
- *Shrink all field views*^⇧K – all field views currently open (whether minimized or not) will be shrunk – i.e. reduced to one quarter of the viewing area and located in the bottom right corner. Only one view will be visible on top of the stack of shrunk views. A shortcut is available in the view's top menubar.
- *Unskrink all field views*^⇧U – all field views currently open (whether minimized or not) will expanded back their their previous size and location. A shortcut is available in the view's top menubar.

- *Cycle forward field views*^⇧↑ – all field views currently open (and not minimized) will be cycled so that each in turn appears in front of the others. This is particularly useful when the field views have been shrunk into a stack at the bottom right corner. A shortcut is available in the view's top menubar.
- *Cycle backward field views*^⇧F↓ – – all field views currently open (and not minimized) will be cycled in reverse order so that each in turn appears in front of the others. This is particularly useful when the field views have been shrunk into a stack at the bottom right corner. A shortcut is available in the view's top menubar.
- *Close all field views*^⇧X – all field views currently open (whether minimized or not) will be closed.
- *Hide toolbars for all field views*^⇧T – all field views currently open (whether minimized or not) will be closed.
- *Play all animated field views* – the user may have open a number of field views that can be animated (i.e. played as a video) and would like to animate these all at the same time. This menu option achieves this goal.
- *Replay all animated field views – this option plays all currently open animated field views in reverse.*
- *Pause all animated field views* – if all animated field views are playing this option will pause/stop them.
- *Go to start of all animated field views* – sets all animated field views to their first frame.
- *Go to end of all animated field views* – sets all animated field views to their final frame.
- *Go to next for all animated field views* – moves all animated field views forward one frame.
- *Go to previous for all animated field views* – moves all animated field views backward one frame.

Manual Menu

- *Open manual …* – allows the user to open a PDF document, termed a manual in the *Streams* system.
- *Select manual directory* – Streams provides four manuals as described in chapter 1. It is likely the user will wish to access those manuals on a regular basis and to ease this task a separate directory, known as the manual directory, is recorded by the system. When the user attempts to open one of these manuals, using one of the following four menu options, the system looks for the manual in the manual directory. If found the manual is opened directly. If not found s dialog box is opened that allows the user to search the file structure for the manual of interest. This menu option allows the user to select the manual directory. This directory is stored in the applications' configuration file and is reinstated each time the program is opened.
- *Open System Theory and Design manual* – this option opens the System Theory and Design manual (in PDF format) located in the manual

directory. If not found in that directory the user is prompted to find the manual in the file system.

- *Open Users Guide: Core Objects manual* – this option opens the User's Guide: Tablet Core manual (in PDF format) located in the manual directory. If not found in that directory the user is prompted to find the manual in the file system.
- *Open Users Guide: Field Objects manual* – this option opens the User's Guide: Field Objects manual (in PDF format) located in the manual directory. If not found in that directory the user is prompted to find the manual in the file system.
- *Open Users Guide: Tablet Objects manual* – this option opens the User's Guide: Tablet Objects manual (in PDF format) located in the manual directory. If not found in that directory the user is prompted to find the manual in the file system.

Tools Menu

- *Display image…* – allows the user to view any image file. A dialog box containing the image is displayed. Text boxes record the position and pixel intensity as the cursor moves over the image. As *Streams* does not know what type of data is stored in the image, the 32bit value at each pixel is translated into an RGB value, a 10 bit gray-scale value, a 12-bit gray-scale value, and a real value. The user can save the image in another format by clicking the Save button. If a TIFF file format is chosen the user has the option of selecting the image resolution (dpi).

Info Menu

- *System info…* – the option displays a dialog box that lists the version of the operating system, the java run time environment and the java virtual machine. This information is probably not useful to most users.
- *About…* – provides information on the software designer and the software version.

Popup menu

A popup menu can be displayed by right clicking the mouse on any portion of the view's background. This popup menu provides a number of the same menu options available through the menu bar. These are

- *Create image sequence…* - see image menu.
- *Open image sequence …* - see image menu.
- *Open particle 2D record …* - see particle menu.
- *Open particle 3D record …* - see particle menu.
- *Open particle 2D bundle record …* - see particle menu.
- *Create velocity-intensity time series …* - see Eulerian menu.
- *Open velocity 2D time series …* - see Eulerian menu.

- *Open velocity 3D time series* … - see Eulerian menu.
- *Open displacement 2D time series* … - see Eulerian menu.
- *Open displacement 3D time series* … - see Eulerian menu.
- *Open displacement 3D surface time series* … - see Eulerian menu.
- *Open intensity time series* … - see Eulerian menu.
- *Open velocity-intensity time series* … - see Eulerian menu.
- *Open density time series* … - see Eulerian menu.
- *Open shape time series* … - see Eulerian menu.
- *Open Lagrangian 2D path field* … - see Lagrangian menu.
- *Open Lagrangian 3D path field* … - see Lagrangian menu.
- *Open tablet* … - see tablet menu.
- *Open process* … - see process menu.
- *Open field* … - see field menu.
- *Minimise all field views* - see field menu.
- *Tile all field views (with the 4 tiling options)* - see field menu.
- *Stack all field views* - see field menu.
- *Shrink all field views* - see field menu.
- *Unshrink all field views* - see field menu.
- *Cycle forward all field views* - see field menu.
- *Cycle backward all field views* - see field menu.
- *Close all field views* - see field menu.
- *Open manual*… - see manual menu.

2.2.2 Object list view

Description

The object list view provides a list of the objects currently opened in *Streams*. These include the image sequences, particle records and particle record bundles, Eulerian fields, Lagrangian fields, tablets, free processes, fields and manuals. The name of each object and its associated icon, together with the name of the file in which it is stored (the file name is not displayed for manuals), is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu. A dashed line is placed between each category of objects. Objects in this view can be dragged and dropped to other views that support such actions. For example, particle 2D records can be dragged and dropped onto the particle view of another particle 2D record, or fields can be dragged and dropped onto tablet analysis views.

Access

The object list view is opened through the *Streams* View menu. It is automatically opened on startup and is positioned at the top left of the application window.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

Popup menu

When a single object in the list is selected and the right mouse button clicked the popup menu for that object will be displayed. However if multiple objects are selected a popup menu can be displayed by right clicking the mouse. This popup menu provides the following self-explanatory options that act on all of the objects selected:

- *Close all views*
- *Close*

2.2.3 Image sequence list view

Description

The image sequence list view provides a list of the image sequences currently opened in *Streams*. The name of each object and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.4 Particle record list view

Description

The particle record list view provides a list of particle records (both 2D and 3D) currently opened in *Streams*. The name of each particle record and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu. This view supports dragging of records to other views.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.5 Particle record bundle list view

Description

The particle record bundle list view provides a list of particle record bundles currently opened in *Streams*. The name of each particle record bundle and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these

objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.6 Eulerian field list view

Description

The Eulerian field list view provides a list of the Eulerian fields currently opened in *Streams*. The name of each Eulerian field and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.7 Tablet list view

Description

The Tablet list view provides a list of the tablets currently opened in *Streams*. The name of each tablet and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.8 Lagrangian field list view

Description

The Lagrangian field list view provides a list of the Lagrangian fields currently opened in *Streams*. The name of each Lagrangian field and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.9 Process list view

Description

The process list view provides a list of the free processes currently opened in *Streams*. The name of each object and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.10 Field list view

Description

The field list view provides a list of the fields currently opened in *Streams*. The name of each field and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu. This view supports dragging of fields to other views.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.11 Manual list view

Description

The manual list view provides a list of the manuals currently opened in *Streams*. The name of each field and its associated icon is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu. This view supports dragging of fields to other views.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.
- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

2.2.12 Diagnostic list view

Description

Internal events that are of interest to the developer are saved as diagnostics within the system. Each diagnostic includes a time and date of occurrence and a message indicating what has occurred. The diagnostic list view displays the currently recorded diagnostics in a table.

Buttons at the bottom of the view enable the user to delete specific diagnostics from the record or to clear all diagnostics.

This view is unlikely to be of interest to the user.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

2.2.13 Field registry list view

Description

The field registry list view provides a list of the fields currently registered in *Streams*. The field registry is described in chapter 1 of the *User's Guide: Field Objects* manual. The name of each field and its associated icon, together with the name of the file in which it is stored, is displayed. The user can interact directly with these objects by double clicking them to open their default view, or right clicking to display their popup menu.

The list of fields supports drag enabling the user to drag these fields to other objects/views that are able to accept a drop action.

Access

This view is opened through the *Streams* View menu. It is not opened on startup.

Menus

View Menu

- *Refresh* – this option will refresh the objects in the view. Normally the user should never have to invoke this refresh manually.

- *Close selected objects* – closes all of the objects currently selected in the list.
- *Close all objects* – closes all objects in the list.

3 Image Sequences

3.1 Description

Image sequences are the starting point for image processing in the *Streams* system. An image sequence represents a complete video record comprising a sequence of still images captured at a fixed frame rate or as pulsed pairs. When the data represented by the images corresponds to two-dimensional data the user typically provides scales that enable the pixels in an image to be converted into physical coordinates.

The image files included in an image sequence may be in a number of image formats. These formats are BMP, JPG, TIF, PNG and GIF. At the same time the data stored within these image files can be of a variety of types. A standard digital video camera typically utilizes one sensor and only a single colour intensity (red, green or blue) is stored for each pixel. The colours stored alternate according the Bayer filter standard and full colour images can be reproduced from this data. More advanced digital video cameras may utilize three sensors each of which records intensity of a single colour intensity for the entire image. These cameras store their colour information in a 32 bit colour format for each pixel that comprises three 8 bit values corresponding to red, green and blue intensities, and an additional 8 bit value corresponding to the alpha setting (transparency), which is not used in *Streams*.

On the other hand, images that have been captured by non-standard scientific digital video cameras may represent 10-bit or 12-bit gray-scale images. Up until, and including, version 3.1 of the software *Streams* specified its own standard for the storage of this gray scale data – namely, each pixel was assumed to store a 32 bit RGB value with the most significant 8 bits in the gray-scale signal being stored in the red channel, with the remaining bits stored in the lowest 2 or 4 bits of the green channel. The blue and alpha channels were ignored in this case. In *Streams* 3.2 and beyond the software has adopted the GigE Vision Streaming Protocol© (GVSP) standard whereby the most significant 8 bits are stored in the red channel while the least significant bits are stored in the most significant bits of the green channel. *Streams* provides a mechanism through the filter pipeline process through which images stored in the previous 10- and 12-bit gray scale formats can be converted to the GVSP standard (see section 23.4.8).

Finally, *Streams* itself may produce images that contain floating point information at each pixel (for example when light intensity is converted to dye concentration). Such images are referred to as storing real values. In this case the IEEE 32 bit floating point representation of the real value is copied directly into the 32 bit colour intensity.

When an image sequence is created the type of data stored in the images must be specified.

All images in an image sequence must have the same dimensions (measured in pixels). This requirement may be checked when the image sequence is created if the user wishes, and any images that do not conform are excluded. This checking process is time consuming as each image must be loaded from disk. Therefore checking is not the default option.

Each image sequence is allocated a name by the user and a textual description can also be provided. The user can specify the temporal and spatial units to be used. The default units are seconds and millimeters respectively.

3.2 Views

The image sequence provides three different views of the data it stores. These are:

- details view
- image view
- process view

3.2.1 Details view

Description

The details view of an image sequence displays the key information about the sequence on three tabs as described below:

***Details*:**

This tab provides core information about the image sequence.

- *Sequence name* – a name chosen by the user to describe the contents of the sequence.

- *Image type* – this combo box specifies the type of data stored in the image – RGB, 10 bit gray scale, 12 gray scale or real values.
- *Number of frames* – this field displays the number of images in the sequence. It cannot be changed directly by the user.
- *Temporal unit* – the temporal unit. The default is seconds.
- *Time step* – this field specifies the time between frames or pairs of frames in the case of an image sequence containing pulsed images.
- *Pulsed time step* – if the image sequence stores pulsed images (i.e. the images are grouped into pairs) this field specifies the time between the images in a pair.
- *Pulsed* – this checkbox should be set if the image sequence contains pulsed images.
- *Spatial unit* – the physical unit to be used when specifying the scales for converting from pixels to physical dimensions.
- *x scale* – the conversion factor in the horizontal direction that will convert pixels to the specified physical unit.
- *y scale* – the conversion factor in the vertical direction that will convert pixels to the specified physical unit.
- *Width (pixs)* – the width of the images in the sequence in pixels. All images must have matching widths and heights. The user cannot edit this field.
- *Height (pixs)* – the height of the images in the sequence in pixels. All images must have matching widths and heights. The user cannot edit this field.

***Description*:**

This tab provides the user with an editable text area in which to provide any textual information about the sequence that they see as appropriate.

- *Description* – the descriptive text.

***Frames*:**

The images in an image sequence are not stored within the object. Instead the image sequence stores references to the locations of its images on disk. This tab lists the images in the image sequence and their location. Image files can be stored according to one of two protocols. The *Absolute* protocol specifies that the image files are stored in a directory that is fixed in the computer's file system. Moving the file that contains the image sequence object will have no impact on the system's ability to access the images. However if the image sequence file and images are moved to another computer or disk then the system will be unable to locate the images. When an image sequence is first created the Absolute protocol is the default. The *Relative* protocol is available when the image files are stored in the same directory as the image sequence itself or in a subdirectory of that directory. Under this protocol the image files are retrieved from a directory that is relatively located in the file system. In this case the image sequence file and its images can be moved

to a different file system and the images can still be retrieved provided their location relative to the image sequence has not changed. The user is able to switch between these two protocols at will.

- *Image files directory* – when the Absolute protocol is set this field displays the full path of the directory in which the image files is stored. When the Relative protocol is specified this field displays the path to the image files starting from the directory in which the image sequence itself is stored. It cannot be edited by the user.
- *File directory protocol* – this combo box specifies the file directory protocol.
- *Image files* – the list box lists the images in the image sequence.
- *Add…* – this button allows the user to select additional images to be added to the image sequence.
- *Insert before…* – this button allows the user to insert additional images before the currently selected image in the list.
- *Delete* – this button allows the user to delete the selected images in the list.
- *Edit list…* – this button displays a dialog box that enables the user to edit the list of images in a simple way. Effectively the user can select the first image in the list that they wish to retain and which of the following images they wish to retain. One image in every N images is kept. The user specifies the number N. For example if N is set to 3 then only every third image is retained.

The user is able to edit many of the parameters specified in the details view. In particular the images in the sequence can be changed after the sequence is created. It is even possible to delete all images from the sequence and to define a new set of images at a later time. Changes made to this view are saved by pressing the Save button. Changes are discarded if the Cancel button is pressed.

Access

This view is accessed through the popup menu of an image sequence. Image sequences are displayed in the *Streams* object and image sequence list views.

3.2.2 Image view

Description

The image view provides the user with an environment in which to inspect the images in an image sequence. The user has the choice as to whether the data in the image should be converted to a viewable format or to leave it in its raw state. For standard RGB images the images are viewable in their raw format. But for gray scale images and images

storing real data this is not the case. If a gray scale image is displayed in its raw format the image will be primarily red and green. If a real value image is displayed in its raw format, whereby the 4 bytes of data in the IEEE format are interpretted as alpha, red, green and blue values, the screen image will generally be uninformative and confusing. By selecting the viewable option (the default setting when the view is opened) gray scale images are converted into standard RGB images where all colours have the same intensity (effectively truncating the 10 or 12 bit data to 8 bit data). For real images the images are converted to 8 bit grayscale images with black corresponding to the smallest real value in the image and white corresponding the largest real value in the image. In both cases the original data is not affected in any way.

The view displays an image from the sequence beneath four toolbars. The toolbars provide the user with a simple video replay facility whereby the images can be played forward or backward, one frame at a time or continuously. It should be noted that the continuous motion replay is unlikely to occur at real speeds due to the time taken for each image to be loaded from disk. However, the user can select a time delay between frames, which can be used to slow down the replay if they wish.

Two of the toolbars provide the user with information about the cursor location and the intensity of the pixel over which the cursor resides. The view interprets the intensity at each pixel according to the data type specified when the sequence was created, and this is unaffected by the user choosing to display the image in a viewable form.

The last toolbar enables the user to explore the output of some of the processes that can act on an image sequence.

Details of the toolbars are given next.

Video toolbar

The top toolbar is the video toolbar. This toolbar provides the video controls (see table 3.1) that allow the user to control the video features of the view.

Table 3.1 Animation controls in the image view of an image sequence.

Controls	Description
Frame displayed	This combo box displays the frame currently displayed in the view. The frame number and time are both shown.
 *Go to next image*	Steps the view forward one frame at a time. If the image has the keyboard focus (achieved by clicking anywhere on the image) pressing the right arrow key causes the same action.
 *Go to previous image*	Steps the view backward one frame at a time. If the image has the keyboard focus (achieved by clicking anywhere on the image) pressing the left arrow key causes the same action.
 *Play sequence*	Plays the sequence of images forwards in time until the last frame is reached. The speed of animation is determined by the setting in the animation time delay field.
 *Replay sequence*	Plays the sequence of images backwards in time until the first frame is reached. The speed of animation is determined by the setting in the animation time delay field.
 *Pause*	Pauses the animation and is only available if one of the play or replay buttons is currently pressed
 *Go to end of sequence*	Moves to the end of the sequence, displaying just the final frame.
 *Go to beginning of sequence*	Moves to the beginning of the sequence displaying the first frame.
Animation time delay (ms)	This text field allows the user to specify the time delay between frames during an animation. This figure is in milliseconds and by default is set to the time step of the image sequence.

Cursor toolbar

The cursor toolbar, second from the top, displays the location of the cursor on screen and indicates the current drawing tool.

- *x* – This text box displays the x coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the image. It can be edited by the user, in which case the cursor is moved to the designated position on the image.
- *y* – This text box displays the y coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the image. It can be edited by the user, in which case the cursor is moved to the designated position on the image.
- *Drawing tool* – This text box displays the name of the currently selected drawing tool. This tool is used for defining regions in the view (see below and chapter 47). The tool can be changed using the *Region* menu.

Pixel toolbar

The pixel toolbar displays the pixel intensities corresponding to the current cursor location (the intensities displayed are unaffected by whether the image is viewed in a viewable format or its raw form).

- *Red* – This text box displays the red intensity of the pixel currently under the cursor if the image stores RGB data. It displays the word "undefined" if the data type is not RGB.
- *Green* – This text box displays the green intensity of the pixel currently under the cursor if the image stores RGB data. It displays the word "undefined" if the data type is not RGB.
- *Blue* – This text box displays the blue intensity of the pixel currently under the cursor if the image stores RGB data. It displays the word "undefined" if the data type is not RGB.
- *Gray (10 bit)* – This text box displays the 10 bit gray scale intensity of the pixel currently under the cursor if the image stores 10 bit gray scale data. It displays the word "undefined" if the data type is not 10 bit gray scale.
- *Gray (12 bit)* – This text box displays the 12 bit gray scale intensity of the pixel currently under the cursor if the image stores 12 bit gray scale data. It displays the word "undefined" if the data type is not 12 bit gray scale.
- *Real* – This text box displays the real value stored at the pixel currently under the cursor if the image stores floating point data. It displays the word "undefined" if the data type is not real.

Image toolbar

The image toolbar sits at the bottom of the toolbar area.

- *Rescale image* – this combo box provides a number of rescaling options ranging from 100% to 5%. A rescaling percentage other than 100% will cause the image to be scaled down to the specified fraction of the

original. Pixel information available in the view will be lost by this rescaling process, although the original image is unaltered.

- *Filters* – this combo box lists all of the filter pipelines currently defined for this image sequence. The user can select any of these in order to display the filtered image produced by the selected pipeline. At the same time the filter view of the selected pipeline is opened so that the user can change parameters in the pipeline filters and observe the effects on the filtered image. To avoid possible ambiguity caused by a number of filter pipelines possessing the same name, each pipeline is preceded by a number indicating its position in the list.
- *PIDs* – this combo box lists all the particle identification processes (PIDs) currently defined for the image sequence. By selecting one of these PIDs the particles that it identifies are overlaid on the image. The PID details view is opened so the user can change parameters dynamically and witness the effect on the particles detected. To avoid possible ambiguity caused by a number of PIDs possessing the same name, each PID is preceded by a number indicating its position in the list.
- *Show particles* – this check box enables the user to easily switch between showing and hiding the particles detected by the selected PID.

The image view is the default view for an image sequence.

Access

This view is accessed by double clicking an image sequence object or using its popup menu.

Menus

File Menu

- *Save filtered image…* – opens a dialog box that allows the user to save the image, currently displayed in the view, to a file. If the user has selected a filter pipeline from the filter pipeline combo box the filtered image will be saved to the file. This provides a short cut for storing a single filtered image from the image sequence.

View Menu

- *Display viewable image* – when selected the image will be converted to a viewable format. In the case of an RGB image this has no effect, but for grayscale and real images it ensures that the displayed image provides a more meaningful visual representation of the data in the image. This is selected by default.
- *Show particles…* – opens a dialog box that displays the details of all particles detected by the current PID (see Image toolbar). This dialog box comprises a table that lists the particles. On each row all particle data available is displayed – e.g. position, size, shape information etc.
- *Draw monochrome particles* – when selected any particles detected by the current PID (see Image toolbar) will be drawn as red particles.
- *Draw colour particles* – when selected any particles detected by the current PID (see Image toolbar) will be drawn in their true colour.

- *Auto open filter and PID views* – when the user is experimenting with various image filters or particle identifiers using the Image toolbar, the details view of the PID or filter view of the filter pipeline will be opened whenever a change in PID or filter is made. This provides the user with rapid access to the filter or PID parameters so changes can be made and their impact observed in the view. When this menu item is not ticked this feature is turned off.

Overlay Menu

- *Choose overlay Eulerian velocity field…* – allows the user to choose an Eulerian velocity 2D field time series to overlay on this view. The field is selected from those fields already open in *Streams*.
- *Clear velocity field…* – this option clears the currently selected overlay velocity field.
- *Display overlay velocity field…* – if selected this causes the velocity vectors of the currently selected overlay field to be displayed on the image.
- *Choose overlay Lagrangian path field…* – allows the user to choose a Lagrangian path field to overlay on this view. The field is selected from those fields already open in *Streams*.
- *Clear Lagrangian field…* – this option clears the currently selected overlay Lagrangian path field.
- *Display overlay Lagrangian field…* – if selected this causes the paths of the currently selected overlay path field to be displayed on the image.
- *Set overlay display parameters…* – opens a dialog box that enables the user to select certain display options. The names of the currently selected velocity and Lagrangian fields are shown (but can't be changed). The size of the velocity vectors and the lengths of the Lagrangian paths (i.e. the number of frames for each path) can be set along with the colours of the vectors and paths.

Region Menu

- *Manage regions…* – opens the regions 2D manager view (see chapter 47).
- *Load regions…*– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn or .rgn2* extension.
- *Save regions as…*– saves all of the currently defined regions in a *.rgn2* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Unselect all regions* – unselects all regions.
- *Select <-> unselect* – unselects all selected regions and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Draw internal rectangle* – sets the current drawing tool to an internal rectangle. This is the default.
- *Draw internal circle* – sets the current drawing tool to an internal circle.
- *Draw internal polygon* – sets the current drawing tool to an internal polygon.
- *Draw internal freehand* – sets the current drawing tool to an internal freehand.
- *Draw external rectangle* – sets the current drawing tool to an external rectangle.
- *Draw external circle* – sets the current drawing tool to an external circle.
- *Draw external polygon* – sets the current drawing tool to an external polygon.

- *Draw external freehand* – sets the current drawing tool to an external freehand.

Popup menu

When the user has drawn an internal rectangle on the view a right click within the rectangle will display a popup menu with the following options:

- *Show intensity matrix – this option displays a dialog box that provides detailed information regarding the pixels in the region selected and any particles that have been identified in this region. See below.*

3.2.2.1 Creating regions

Regions (see chapter 47) are two-dimensional shapes that can be used for selecting areas within an image. Regions are created within the image sequence image view by using a range of drawing tools selectable within the *Region* menu of the view. Four different 2D geometries are available – rectangle, circle, polygon and freehand drawing (in essence a high order polygon). For each geometry the region inside or outside the figure can be defined as the region of interest. The default drawing tool is the Internal Rectangle. Thus the region is defined as the area within a rectangle drawn on screen. If External Rectangle was selected as the drawing tool then the region would be defined to be the area outside the rectangle. The colour of the region's outline distinguishes between the internal and external setting. Internal regions are drawn in green or blue, and external regions are drawn in orange or red.

Details regarding how the different drawing tools are used are provided in chapter 44, together with information on the manipulation of regions already created. It is important to note that the image display area must have the mouse and keyboard focus during region manipulation. A small rectangular indicator in the bottom right hand corner of the view is green when the area has the focus and red when it does not. To regain the focus simply click the mouse in the image display area.

Regions have a number of uses, for example in the definition of certain filters that are based on regions and in the creation of intensity fields.

3.2.2.2 Intensity matrix dialog

The intensity matrix dialog provides a mechanism for investigating the pixel information in a user defined rectangular region of the image. The dialog box displays the pixel intensities on a rectangular grid, using, in

general, false colours to indicate the intensities. At the top of the dialog box are three toolbars.

Display toolbar

The topmost toolbar is the display toolbar.

- *Grid* – when selected a grid of lines is displayed to indicate the boundaries between pixels.
- *Particles* – when selected any particles that have been identified by a PID in the original image will be displayed with a black dot at the centre and a black circle to indicate the extent of the particle.

Cursor toolbar

The cursor toolbar is the middle toolbar.

- *x* – this text field displays the current x position of the cursor.
- *y* – this text box displays the current y position of the cursor.
- *Intensity* – this text box displays the intensity at the current cursor location. The choice of intensity displayed is set by the user through the View menu. For example the user may choose to display the red intensity in an RGB image, or the RGB intensity which is the average of the three colours. The one exception is when the user chooses to display the true RGB image. In this case the RGB intensity is displayed in the text field.

Colour toolbar

The image toolbar sits at the bottom of the toolbar area.

- *Min colour value* – this text box sets the minimum intensity in the false colour display. This is set automatically as the lowest intensity in the image. If the user changes this to a higher value intensities less than the value specified will be displayed in black.
- *Max colour value* – this text box sets the maximum intensity in the false colour display. This is set automatically as the highest intensity in the image. If the user changes this to a lower value intensities greater than the value specified will be displayed in black.

Menus

File Menu

- *Save as image…* – opens a dialog box enables to save the false colour image to a file. The user specifies the resolution of the image file in pixels and the file name and location.

View Menu

- *Show true image* – when selected the true image will be displayed assuming that the pixel intensity stores RGB information. The intensity displayed in the intensity text field in the Cursor toolbar will be the average intensity of the 3 colours.
- *Show RGB intensity* – when selected the average RGB intensity (i.e. the average of the red, green and blue guns) will be displayed.
- *Show red* – when selected the red intensity will be displayed.
- *Show green* – when selected the green intensity will be displayed.
- *Show blue* – when selected the blue intensity will be displayed.
- *Show 10 bit gray scale* – when selected the pixel data will be interpreted as corresponding to a 10 bit gray scale image this gray scale intensity will be displayed. Intensities will range from 0 to 1023.
- *Show 12 bit gray scale* – when selected the pixel data will be interpreted as corresponding to a 12 bit gray scale image this gray scale intensity will be displayed. Intensities will range from 0 to 4095.
- *Show real value* – when selected the pixel data will be interpreted as corresponding to real values and these real value will be displayed.
- *Show grid* – displays the grid. This is identical to the option in the toolbar.
- *Show particles* – displays the particle. This is identical to the option in the toolbar.
- *Change decimal places…* – this displays a dialog box that allows the user to change the number of decimal places used to display the spatial coordinates on the axes.
- *Hide toolbars* – this option hides the toolbars at the top of the display, thus providing more display real estate for the image itself.

Colour Menu

- *Use false colour* – when selected the pixel intensities are displayed as false colours. The colours used can be selected by the user (see below). The default colour scheme runs from blue for the lowest intensity through green, yellow, orange and up to red for the highest intensity. If this option is not selected the intensities will be displayed using a gray scale, with black for the lowest intensity and white for the highest.
- *Change grid colour…*– this allows the user to change the colour of the grid.
- *Change axes colour…*– this allows the user to change the colour of the axes.
- *Change label colour…* – this allows the user to change the colour of the labels.
- *Change false colour scheme…* – this option opens a dialog box that enables the user to customize the colour scheme used to specify the intensities.

Particle Menu

- *Show particle list…* – opens a dialog box that lists the particles and their properties identified in this subimage.

Popup menu

A right click within the image will display a popup menu with the following options:

- *Show grid* – displays the grid. This is identical to the option in the toolbar.
- *Show particles* – displays the particle. This is identical to the option in the toolbar.

3.2.3 Process view

Description

The process view provides access to an image sequence's processes. All entities that support processes will have an identical view. This view will be described in detail here for a general entity and other entities will refer to this section.

The process view provides two list boxes, each of which has a number of buttons beneath it.

Processes

The top list box contains the currently defined processes. The names and characterizing icons are shown for each process together with their input and output entity types. This list box supports drag and drop of processes so that processes defined for one entity can be easily copied to another. The processes in the list box are colour coded in the following way.

- *Black* – the process is a primary process for the entity and is ready to execute. A process that is displayed in black can be executed immediately through its popup menu.
- *Blue* – the process is a primary process but currently is not properly defined and therefore cannot execute in its current state.
- *Red* – the process is a secondary process and cannot process the entity.

There are six buttons beneath the process list. These are

- *New* – displays a dialog box from which the user can select a new process. The dialog box presents two lists, the one on the left contains the primary processes for the entity and the one on the right the secondary processes. The user can select the desired process by clicking on it and choosing OK or simply double clicking on the process. In general when a new process is selected from the box a process with default settings is created and added to the list. The user should then open an appropriate view of the process to set its parameters.
- *Load* – loads a previously defined bound process from disk. Bound processes have the file extension .pcb.
- *Delete* – deletes the currently selected process. The user is asked to confirm the action.
- *Clear* – deletes all currently defined processes. The user is prompted to confirm this action.
- *Add to pipeline* – adds the currently selected process to the end of the process pipeline. If a process is already a component in the pipeline it cannot be added again.
- *Insert in pipeline* – inserts the selected process before the currently selected (highlighted) process in the pipeline. Again, a process cannot be added more than once.

Process pipeline

The bottom list box contains the currently defined pipeline processes. The names and characterizing icons are shown for each process together with their input and output entity types. The processes are colour coded in the following way.

- *Black* – the processes up to and including this process are valid in that their input and output entity types match.
- *Blue* – the process is a valid element in the pipeline but currently is not properly defined and cannot execute.
- *Red* – the process, or one before it in the pipeline, is incompatible with the preceding process due to an incompatibility of input and output entity types.

There are three buttons beneath the pipeline process list. These are

- *Delete* – deletes the currently selected process from the pipeline. If this process is present in the process list it is not deleted from that list. The user is asked to confirm the action.
- *Clear* – deletes all currently defined processes from the pipeline. This action does not remove them from the list of processes. The user is prompted to confirm this action.
- *Execute* – this executes the pipeline. This operation will not be successful unless all processes are displayed in black.

Access

This view is accessed through the popup menu of an image sequence.

3.3 Popup menu

The image sequence provides the following options in its popup menu.

- *Save as…* – saves the image sequence in a new file. All image sequence files have the extension *.seq*. The new file name is displayed next to the image sequence name in the *Streams* object and image sequence list views.
- *Save* – saves the image sequence to disk. If the image sequence currently has no associated file then the user is prompted for a file name. All image sequence files have the extension .seq.
- *Change name…* – allows the user to change the name of the image sequence. This name change will be reflected in the *Streams* image sequence list view. This action can also be performed through the details view.
- *Edit description…* – allows the user to change the description of the image sequence. This action can also be performed through the details view.
- *Delete frames…* – this option allows the user to delete frames (images) from an image sequence. The first and last frames to be deleted are selectable by the user. The frames are renumbered after the deletion if

appropriate. This action is drastic and the user is prompted for confirmation. All views are closed when this action is taken.

- *Open details view* – opens the details view.
- *Open image view* – opens the image view. This is the default view.
- *Open process view* – opens the process view.
- *Execute process pipeline…* – executes the current process pipeline. This will fail unless all processes are valid and the pipeline has no incompatibilities.
- *Close all views* – closes all open views for this object.
- *Close* – removes the image sequence from *Streams*. All views are closed. If the image sequence has unsaved data then the user is prompted to save this data before closure.

3.4 Creating image sequences

Image sequences are created by selecting one of two options in the Image menu of the *Streams* application view. The Create image sequence option creates a sequence using images of standard file format (JPEG etc.) while the Create image sequence from JAI images creates a sequence from images stored using JAI©'s proprietary format (.JAI).

3.4.1 Using standard images

A dialog box is displayed that prompts the user for key information about the image sequence. This dialog box possesses three tabs that closely match those in the sequence's details view. These are:

Details

The user must supply the following image sequence details.

- *Image sequence name* – the sequence name, used to differentiate the sequence from all others in the *Streams* system. In reality Streams will allow two sequences to have the same name but for obvious reasons this should be avoided.
- *Image type* – the type of data stored in the images. This can be RGB colour, gray scale (10bit), gray scale (12 bit), or real values.
- *Temporal unit* – a text string that designates the temporal unit for this image sequence. This unit is used in a number of views and is passed to derived quantities such as particle records. The default is s (seconds).
- *Time step(s)* – the time interval, in seconds, between images in the sequence. This must be the same for all frames. If the sequence is pulsed this is the time step between each pair of images.

- *Pulsed time step (s)* – the time interval, in seconds, between images in a pair if the sequence is pulsed. This cannot be set unless the Pulsed checkbox is selected.
- *Pulsed* – if this checkbox is ticked then the images in the sequence come in pulsed pairs, and two time steps must be specified.
- *Spatial unit* – a text string that designates the spatial unit for this image sequence. This unit is used in a number of views and is passed to derived quantities such as particle records. The default is mm (millimetres).
- *x scale* – the factor used to convert from pixel space to physical space in the x direction. The units are mm/pixel.
- *y scale* – the factor used to convert from pixel space to physical space in the y direction. The units are mm/pixel.
- *Width* – this field cannot be edited by the user and displays a value of –1 until the image files have been specified. In this case the width (in pixels) of the first image is displayed here.
- *Height* – this field cannot be edited by the user and displays a value of –1 until the image files have been specified. In this case the height (in pixels) of the first image is displayed here.
- *Check images* – if this checkbox is ticked then each image in the sequence is checked to ensure that it is of the same dimensions as the first image in the list. While this check guarantees that the image sequence will be valid it is a time consuming process as each image must be loaded in order to be checked. It is suggested that if the integrity of the selected images (i.e. they are of the same size and type) is assured then this checking is unnecessary.

Description

The user can provide a textual description of the image sequence in this tab.

Frames

In this tab the user chooses the files from which the sequence is to be created. Each file corresponds to a frame in the sequence, and they must be listed in the same order as the frames will appear in the sequence. The files must be JPEG (JPG), GIF, PNG, BMP or TIF and all files must reside in the same directory. The names of the selected files are displayed in a scrollable list box, while the directory in which they reside is displayed in the Image files directory field at the top of the tab.

The elements on this tab are:

- *Image files directory* – this field displays the full path of the directory in which the image files is stored. It cannot be edited by the user.
- *File directory protocol* – This combo box specifies the file directory protocol. At creation this is set to Absolute and cannot be changed (see section 3.2.1).

- *Image files* – The list box lists the images in the image sequence.
- *Add…* – This button allows the user to select additional images to be added to the image sequence.
- *Insert before…* – This button allows the user to insert additional images before the currently selected image in the list.
- *Delete* – This button allows the user to delete the selected images in the list.
- *Edit list…* – This button displays a dialog box that enables the user to edit the list of images in a simple way. Effectively the user can select the first image in the list that they wish to retain and which of the following images they wish to retain. One image in every N images is kept. The user specifies the number N. For example if N is set to 3 then only every third image is retained.

The user can display one of the images before creating the image sequence by double clicking on the file name.

Once the user presses the OK button to create the image sequence the images in the selected files are checked for dimensional consistency with the first image in the list if the user has chosen this option. A monitor dialog box is displayed showing the progress of this integrity check. Any images that do not conform to the dimensions of the first image are excluded from the image sequence and the monitor displays that a bad image has been identified and sounds a beep to alert the user.

3.4.2 Using JAI images

The creation of an image sequence from JAI images includes an additional step to that using standard images. This step involves loading each image in the sequence and storing it in one of the standard formats. The resulting image sequence refers to these processed image files, not the original JAI files. Thus *Streams* provides no direct support for image sequences of JAI images other than through this conversion process on creation. All image sequences refer to images stored in one of the standard formats.

A dialog box is displayed that prompts the user for key information about the image sequence. This dialog box possesses three tabs that closely match those in the sequence's details view. These are:

Details

The user must supply the following image sequence details.

- *Image sequence name* – the sequence name, used to differentiate the sequence from all others in the *Streams* system. In reality Streams will

allow two sequences to have the same name but for obvious reasons this should be avoided.

- *Image type* – the type of data stored in the images. This is determined from the header data in the first JAI file in the list. This cannot be edited by the user.
- *Pixel depth* – the number of bits used to store the image data. This could be 8 bit, 10 bit or 12 bit and is determined from the header data in the first JAI file in the list. This cannot be edited by the user.
- *Number of frames* – The number of images in the image sequence. This cannot be edited by the user.
- *Temporal unit* – a text string that designates the temporal unit for this image sequence. This unit is used in a number of views and is passed to derived quantities such as particle records. The default is s (seconds).
- *Time step(s)* – the time interval, in seconds, between images in the sequence. This must be the same for all frames. If the sequence is pulsed this is the time step between each pair of images.
- *Pulsed time step (s)* – the time interval, in seconds, between images in a pair if the sequence is pulsed. This cannot be set unless the Pulsed checkbox is selected.
- *Pulsed* – if this checkbox is ticked then the images in the sequence come in pulsed pairs, and two time steps must be specified.
- *Spatial unit* – a text string that designates the spatial unit for this image sequence. This unit is used in a number of views and is passed to derived quantities such as particle records. The default is mm (millimetres).
- *x scale* – the factor used to convert from pixel space to physical space in the x direction. The units are mm/pixel.
- *y scale* – the factor used to convert from pixel space to physical space in the y direction. The units are mm/pixel.
- *Width* – this field cannot be edited by the user and displays a value of –1 until the image files have been specified. In this case the width (in pixels) of the first image is displayed here.
- *Height* – this field cannot be edited by the user and displays a value of –1 until the image files have been specified. In this case the height (in pixels) of the first image is displayed here.

Description

The user can provide a textual description of the image sequence in this tab.

Frames

In this tab the user chooses the files from which the sequence is to be created. Each file corresponds to a frame in the sequence, and they must be listed in the same order as the frames will appear in the sequence. The

files must be JAI and all files must reside in the same directory. The names of the selected files are displayed in a scrollable list box.

These images will be read in at the time of image sequence creation and converted to another format. Thus the user is prompted to provide an output file name and location. The selected name will have a 5 digit number added to it when the converted files are produced and saved. The output file extension can be any one of the standard image file formats – JPEG etc.

The elements on this tab are:

- *Output image file name* – this field displays the full path of the file name that will be used for the converted files. This is best specified using the Browse button. It is recommended that file names have the form XXXX_.EXT where EXT can be JPEG, BMP etc. The converted files will then have names XXXX_00000.EXT, XXXX_00001.EXT etc.
- *Browse* – This button allows the user to navigate the directory structure when specifying the output file name (and extension) for the converted files.
- *Image files* – The list box lists the images in the image sequence.
- *Add…* – This button allows the user to select additional images to be added to the image sequence.
- *Insert before…* – This button allows the user to insert additional images before the currently selected image in the list.
- *Delete* – This button allows the user to delete the selected images in the list.
- *Edit list…* – This button displays a dialog box that enables the user to edit the list of images in a simple way. Effectively the user can select the first image in the list that they wish to retain and which of the following images they wish to retain. One image in every N images is kept. The user specifies the number N. For example if N is set to 3 then only every third image is retained.

Once the user presses the OK button to create the image sequence the images in the selected files are loaded sequentially, converted to the specified image format, and saved to the output directory with the specified numbered file name. A monitor dialog box is displayed showing the progress of this process.

4 Particles in 2D

4.1 Description

A **particle 2D** object represents a particle identified in a digital image, or created artificially in a theoretical model. Each particle is assigned a unique number (starting at 0) in the particle frame in which it appears. The data stored with the particle is dependent on two user choices. The following data is always stored with the particle:

- *x, y* – the position of the particle. The units for these and other spatial quantities is dependent on the spatial unit specified in the associated particle 2D record (see chapter 6).
- *radius* – the equivalent radius of the particle. This is computed from the calculated area of the particle in the original image assuming the particle is circular.
- *colour* – red, green and blue intensities of the particle each in the range 0-255 even if the original image has higher resolution. All three intensities are the same for a monochrome image.

The following data is stored if the user chooses to store shape information.

- *Ix, Iy* – the moments of inertia of the particle about its principal axes (assuming the particle is a flat plate). These two values are normally replaced by the aspect ratio that is computed from these moments as if the particle was an ellipse.
- *theta* – the angle the major principal axis of the particle makes with the x axis. This provides orientation information about the particle again assuming it is an ellipse.
- *eccentricity* – the eccentricity of the particle. This is a measure of its irregularity calculated as the normalized distance between the centre of mass of the particle and the centre of the smallest rectangle that encloses the particle. If the particle is a perfect circle or ellipse the eccentricity will be 0. The distance is normalized by the length of the major axis.

The following data is stored if the user chooses to store the particle intensity matrix.

- *Intensity matrix* – this is a two-dimensional matrix containing the intensity (average of the three colour values) at each pixel within the smallest rectangle that contains the particle. The intensities of pixels with non-zero intensities, that are not part of the particle, are set to zero. This matrix typically requires considerable memory and should only be stored if the information is crucial to the application and the total number of particles is relatively small.

Particles are generally used in particle tracking analysis. Thus they also store the particles to which they are matched in the previous and next frames. Until a PTV 2D analysis creates these matches they are undefined. Particles that are created from theoretical models also know which particles in the previous and next frames are their correct, or true, matches. It is possible for the user to artificially create true matches for particles created from digital images (see chapter 6) using the current set of particle matches. In this case such particles are treated as if they were derived from a theoretical model.

Particles created in theoretical models do not store shape data nor intensity matrices.

4.2 Views

A particle object has two views.

- *Details view.*
- *Intensity matrix view.*

4.2.1 Details view

Description

The details view displays the data, as described above, pertaining to a particle.

Access

It is accessed through

- the particle list view of a particle 2D record.
- the particle matches view of a particle 2D frame.

4.2.2 Intensity matrix view

Description

The intensity matrix view displays the rectangular intensity matrix that contains the particle. The user has some control over how the matrix is displayed. For example whether an artificial colour scale is used or a gray scale.

Access

It is accessed through

- the particle matches view of a particle 2D frame.

5 Particle Frames in 2D

5.1 Description

A **particle 2D frame** object represents a frame of particles extracted from a video record, or produced by a theoretical model. It contains a list of particles identified in the digital image, or artificially produced by the theoretical model. A particle frame created from a digital image keeps track of the image file from which it was created. This file is known as its source.

Each particle frame is part of a particle 2D record and is assigned a unique number (starting at 0) within that record.

5.2 Views

The particle 2D frame provides four views. These are

- details view
- particle list view
- matches view
- segmentation view

5.2.1 Details view

Description

This view displays details of the particle frame, including the number of particles within it. The details displayed vary depending on whether the particles within the frame have true matches recorded or not (i.e. have been created by a theoretical model or have had true matches created artificially). For a frame with true matches, details about the number of correct and incorrect matches are included.

Access

This view is accessed through the popup menu of a particle 2D frame. Frames for a record are displayed in a particle 2D record details view.

5.2.2 Particle list view

Description

This view displays a table of the particles within the frame. For each particle all of the available particle information is tabulated – position, radius, colour (RGB), area, aspect ratio, etc.

Access

This view is accessed through the popup menu of a particle 2D frame. Frames for a record are displayed in a particle 2D record details view.

Menus

Sort Menu

- *By number…* – sorts the table by particle number.
- *By x…* – sorts the table by x.
- *By y…* – sorts the table by y.
- *By radius…* – sorts the table by particle radius.
- *By aspect ratio…* – sorts the table by particle aspect ratio.
- *By intensity…* – sorts the table by particle intensity.

5.2.3 Matches view

Description

The matches view for a particle frame comprises a table giving particle matching information. The particles in the frame are listed in numerical order and their key statistics (location, radius and colour) are included with them. In the second and third columns the particles that these particles are matched to in the previous and next frames are displayed along with their positions. If the particles in the frame have true match information then the particle numbers of the true matches are also included in these columns. In the final column the path length of each particle is listed. A path length of 1 indicates that the particle is not matched to particles in either the previous or next frames.

This is the default view for a particle 2D frame.

Access

This view is accessed through the popup menu of a particle 2D frame or by double clicking the frame. Frames for a particle 2D record are displayed in a particle 2D record details view.

Menus

Particle Menu

- *Display particle details…* – displays the details view of the selected particle.

5.2.4 Segmentation view

Description

Many of the processes involving particle 2D records, and hence particle 2D frames, are expedited if particles can be rapidly located in the frame. **Segmentation** of particle frames is a mechanism for doing this. A **segment** is a rectangular region within a particle frame that records the particles that lie within the region. The segmentation of a particle frame comprises a rectangular mesh of segments that cover the entire frame. A segmentation is created through the popup menu of the particle 2D record and at that time the number segments in each row and column of the segmentation can be defined.

The segmentation of a particle frame will provide substantial speed improvements in a range of particle tracking velocimetry processes – for example finding candidate particles for matching – but it does carry with it an extra storage overhead, typically 20%.

The segmentation view provides a list of the segments in the frame's segmentation and for each segment it lists the particles in that segment.

Access

This view is accessed through the popup menu of a particle 2D frame. Frames for a particle 2D record are displayed in a particle 2D record details view.

5.3 Popup menu

The popup menu is displayed by right clicking on the particle 2D frame object when it is displayed in the particle 2D record details view. The menu options are

- *Save particle details to text file…* – saves all particle information to a comma (,) delimited text file selected by the user.
- *Open details view* – opens the details view of the frame.
- *Open particle list view* – opens the particle list view of the frame.
- *Open matches view* – opens the matches view of the frame.
- *Open segmentation view* – opens the segmentation view of the frame.

6 Particle Records in 2D

6.1 Description

The particle 2D record is a key object in the *Streams* system. A particle 2D record represents a sequence of frames containing two-dimensional particles obtained, either from a sequence of digital images, or from a theoretical model. It contains a list of particle 2D frames (see chapter 5) each of which, in turn, contains the particles that have been identified in that frame. The record needs to store a number of basic experimental parameters such as the time step between frames and the size of the spatial domain that contains all of the particles within the record.

Particle records can be pulsed or not pulsed. Pulsed records have particle frames that occur in pairs, where the time step between pairs of frames is typically large compared to the time between frames within a pair. Records that are not pulsed have a sequence of frames that have a constant time step between them.

Each particle record is allocated a name by the user and a textual description can also be provided.

The segmentation of particle records, as discussed in section 5.2.4, can lead to substantial reductions in computation time for certain PTV operations. While segmentation carries a modest storage overhead, and a small computational expense (typically a few seconds to create the segmentation), the execution speed improvements can be very significant – more than an order of magnitude in some cases – and therefore the segmenting of particle records should be adopted as standard practice if no memory issues exist. Particle records are segmented by default.

6.2 Views

The particle record provides ten different views of the data it stores. These are:

- details view
- particle view
- process view
- match summary view
- comparison view
- diagnostic view

- particle details view
- match details view
- candidate matches view
- particle path view

6.2.1 Details View

Description

The details view for a particle record displays key information about the record, including its size, spatial domain, time step, segmentation etc.

The details view allows the user to change a number of the record parameters. This includes the name of the record, the description, the temporal and spatial units and the time step between frames. The user must press the Save button for these changes to take effect. If the changes are to be discarded press the Cancel button.

The view has four tabs, Parameters, Description, Frames and Segmentation. These are described below.

Parameters

- *Record name* – the name associated with the record. This can be changed in the details view or through the object's popup menu. Note that this is not the name of the file in which the record is stored, although of course the user could choose to make them the same.
- *Record type* – this field is not editable. It specifies whether the record was generated from a model or from images.
- *True matches exist* – this box is ticked when true matches exist for this record. It cannot be edited.
- *Temporal unit* – the user defined time unit. The default is s (seconds).
- *Time step (unit)* – the time step between frames in the units defined above.
- *Pulsed time step (unit)* – when the record is a pulsed record this is the time step between frames within a pair. When not pulsed it has a value of -1.
- *Number of frames* – the user number of frames in the record. This is not editable by the user.
- *Spatial unit* – the user defined space unit. The default is mm (millimetres).
- *x min (unit)* – the minimum value of x for the record. All particles have an x coordinate equal to or larger then this. This is not editable.
- *x max (unit)* – the maximum value of x for the record. All particles have an x coordinate equal to or less then this. This is not editable.

- *y min (unit)* – the minimum value of y for the record. All particles have an y coordinate equal to or larger then this. This is not editable.
- *y max (unit)* – the maximum value of y for the record. All particles have an y coordinate equal to or less then this. This is not editable.

Description

- *Description* – the user can provide a description of the record in the text box provided.

Frames

- *Frames* – a list of the particle 3D record frames contained in the record is provided. The user can access the popup menus of the frames in this view and hence get access to particle information through the frame views.

Segmentation

- *Segments are used* – this box is ticked if segmentation of the record has occurred. This cannot be changed in the view but can be changed through the popup menu of the record.
- *Number of x segments* – the number of segments used to divide the x axis. This is not editable.
- *Number of y segments* – the number of segments used to divide the y axis. This is not editable.

Access

This view is accessed through the popup menu of a particle 2D record. Particle 2D records are displayed in the *Streams* object and particle record list views.

6.2.2 Particle view

Description

The particle view is the most useful of the particle record views, and it gives rise to four more views – the particle details, the match details, candidate matches, and particle path views. It displays an idealised picture of the particle record, representing particles in each frame as points on the screen. The user can overlay as many frames as they wish, thus effectively obtaining a long exposure photograph of the flow that provides a strong sense of the flow patterns. This idealised flow view can be animated, or played forward or backward one frame at a time, just

as if it were a video. This animation can be performed with many frames displayed simultaneously.

Another particle 2D record can be dropped onto this view. If this operation is performed without the ALT/OPTION key pressed the particles in the dropped record will be overlaid on those in the original record – thus all particles in frame 0 of the dropped record will be added to frame 0 of the original record and so on. No checking for compatibility of the two records is undertaken. If the ALT/OPTION key is pressed during this operation the frames of the dropped record are concatenated to the original record – thus if the original record has N frames the frames of the dropped record will become frames N+1 and beyond. The first of these operations cannot be reversed so the user should always make a copy of the original record before undertaking this action. The second action can be reversed by simply deleting the new frames. In both cases the user is asked to confirm the action before it is executed. The dropped record is unaffected by either of these operations.

The view is initially displayed so that it has an optimal size with minimal space wasted.

The view has four toolbars displayed above the particle display area. If the user double clicks the mouse in the blank space between the toolbars the view will resize so that it is of optimal size with minimal waste space.

The toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls (see table 6.1) that allow the user to control the animation features of the view. The buttons provided in this toolbar are common to all views with animation within the *Streams* environment.

Table 6.1 Animation controls in the particle view of a particle record 2D.

	Description
First frame displayed	This combo box displays the first frame currently displayed in the view. The frame number and time are both shown.
Number of frames displayed	This combo box displays the number of frames currently shown.
 *Go to next frame*	Steps the view forward one frame. The number of frames displayed remains the same unless that number of frames does not exist before the end of the record is reached.
 Go to previous frame	Steps the view backward one frame. The number of frames displayed remains the same unless that number of frames does not exist before the end of the record is reached.
 *Play record*	Plays the record forwards in time until the last frame is reached. The speed of animation is determined by the setting in the animation time delay field.
 *Replay record*	Plays the record backwards in time until the first frame is reached. The speed of animation is determined by the setting in the animation time delay field.
 *Pause record*	Pauses the animation and is only available if one of the play or replay buttons is currently pressed.
 *Go to end of record*	Moves to the end of the record, displaying just the final frame.
 *Go to beginning of record*	Moves to the beginning of the record displaying the number of frames specified in the *Number of frames displayed* box.
Animation time delay (ms)	This field allows the user to specify the time delay between frames during an animation. It is in milliseconds and by default is the time step of the record.

Particle toolbar

The particle tool bar sits below the animation toolbar. It allows the user to select the particle and particle matching information that is displayed in the view. The buttons present in this toolbar will depend on whether the particle record has true matches information.

- *Particles* – when this button is pressed all the particles in the visible frames are displayed.
- *Numbers* – when this button is pressed all the particle numbers in the visible frames are displayed.
- *PTV* – when this button is pressed all matches between particles in the visible frames (generated by a PTV 2D analysis) are drawn as lines between the particles. By default these matches are drawn in light blue. This colour can be changed through the *Colour* menu.
- *True* – (Only visible if true matches information is available). When this button is pressed the true matches between particles in the visible frames are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 6.2.2.1).
- *Good* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames that match the true matches are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 6.2.2.1).
- *Bad* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames for which different true matches exist (ie the PTV matches are not the correct matches) are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 6.2.2.1).
- *Missed* – (Only visible if true matches information is available). When this button is pressed the true matches between particles in the visible frames that have been missed by a PTV analysis are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 6.2.2.1).
- *False* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames that have no corresponding true match are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 6.2.2.1).
- *Paths* – When this button is pressed the particle paths generated by a PTV analysis, that include the particles in the first visible frame, are drawn as coloured lines. The particles present in these paths do not need to be currently displayed. Each pathline is given a random colour that will change if the user moves through the frames.
- *Path* – When this button is pressed the particle path that includes the particle in the first visible frame whose number matches that specified in the particle number text box is drawn. The colour can be changed in the view layout dialog box (see section 6.2.2.1).

Cursor toolbar

The cursor toolbar, second from the bottom, displays the location of the cursor on screen and indicates the current drawing tool.

- *x* – This text box displays the x coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *y* – This text box displays the y coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *Drawing tool* – This text box displays the name of the currently selected drawing tool. This tool is used for defining regions in the view (see below and chapter 47). The tool can be changed through the *Region* menu.

View toolbar

The view toolbar sits at the bottom of the toolbar area. Here the user can select the portion of the frame that they wish to display. The zoom capability, described below, provides an alternative technique for changing the region of the frame that is displayed.

- *Min x axis* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x and y directions have the same length on screen.
- Max x axis – This text box displays the maximum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Min y axis – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Max y axis – This text box displays the maximum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.

The particle view is the default view for a particle 2D record.

Access

This view is accessed through the popup menu of a particle 2D record or by double clicking the object.

Menus

File Menu

- *Save as image*… – allows the user to save the particle view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user can choose the image size (in pixels) and is able to preview the image before it is stored on disk. Further details of this process are provided in the *User's Guide: Field Objects* manual.
- *Save as image sequence*… allows the user to save each of the frames in the particle record as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the *User's Guide: Field Objects* manual.

View Menu

- *Set layout*… – this displays a dialog box that enables the user to set a range of display parameters common to all particle record particle views (i.e. changing these settings for this view changes them for all particle views). Details regarding this dialog box are provided in section 6.2.2.1).
- *Draw point particles*… – a check menu item that determines whether the particles are displayed as points.
- *Draw true particles*… – a check menu item that determines whether the particles are displayed as their true shapes and sizes.
- *Draw monochrome particles* – if selected the particles are drawn in a single colour (which is set in the view layout – see section 6.2.2.1).
- *Draw true coloured particles* – if selected the particles are drawn in their true colour.
- *Draw colour coded particles*… – if selected the particles are drawn in a single colour except for the first and last frames displayed. Particles in these frames are coloured green and red respectively.
- *Display subset of particles...* – a dialog box is displayed that allows the user to select the particles to display – 1 particle in every N particles is displayed and the user specifies N.
- *Show particles* – a check menu item that determines whether the particles are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show particle numbers* – if selected the particle numbers are displayed alongside the particles in the view. This tends to useful only with modest numbers of particles or when the user has zoomed in on a portion of the flow.

- *Show PTV matches* – a check menu item that determines whether the PTV matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show true matches* – (Only visible if true match information is available) a check menu item that determines whether the true matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show good matches* – (Only visible if true match information is available) a check menu item that determines whether the good matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show bad matches* – (Only visible if true match information is available) a check menu item that determines whether the bad matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show missed matches* – (Only visible if true match information is available) a check menu item that determines whether the missed matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show false starts …* – (Only visible if true match information is available) a check menu item that determines whether the false starts are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show particle paths – a check menu item that determines whether the particle paths are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show triangulation – when selected the triangulation used for velocity field calculation for the first frame is displayed in the view.
- Show nearest neighbours – when selected the nearest neighbours are displayed in the view. The parameters for determining the nearest neighbours are selected using the Set nearest neighbours parameters… item in the view menu. The point of interest is marked by a red circle and the nearest particles are highlighted by green circles.
- Show info – this item displays the temporal information that is located beneath the x axis.
- Show regions – when selected the currently defined regions are displayed.
- *Redraw during resize* – if set, the view redraws during a resize operation. If the drawing operation is extensive, for example many frames with large numbers of particles, this might cause a lack of responsiveness. If not set redrawing does not occur and the user needs to perform some manual operation after the resize in order to force a redraw.
- Change variable labels… – this item opens a dialog box in which the labels for the x, and y variables can be set. Selecting and right clicking the text in the text boxes enables the user to select Greek or Latin characters.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y and t variables can be set. Sensible minimum values are set for these.

- Set nearest neighbours parameters… – this item opens a dialog box in which the user can specify the parameters used when displaying the nearest particles (neighbours) to a point. The user is prompted for the x and y coordinates of the point of interest along with the number of nearest neighbours to display. A number of the interpolation algorithms employed to compute Eulerian fields, such as the velocity field time series, utilize the nearest particles to a grid point to interpolate field information at that point.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar.
- Adjust view size – this resizes the view window to minimize waste space. The same effect can be achieved by double clicking in the blank space on either side of the toolbars (assuming that they are visible).

Animation Menu

- *Go to next time* – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to previous time* – steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Play* – starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Region Menu

- *Manage regions*… – displays region 2D manager view (see chapter 47).
- *Load regions*…– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn* or a *.rgn2* extension.
- *Save regions as*…– saves all of the currently defined regions in a *.rgn2* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.

- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Draw internal rectangle* – sets the current drawing tool to an internal rectangle. This is the default.
- *Draw internal circle* – sets the current drawing tool to an internal circle.
- *Draw internal polygon* – sets the current drawing tool to an internal polygon.
- *Draw internal freehand* – sets the current drawing tool to an internal freehand.
- *Draw external rectangle* – sets the current drawing tool to an external rectangle.
- *Draw external circle* – sets the current drawing tool to an external circle.
- *Draw external polygon* – sets the current drawing tool to an external polygon.
- *Draw external freehand* – sets the current drawing tool to an external freehand.
- *Delete particles in selected regions…* – on user confirmation this option deletes all particles, in the currently displayed frames, that are located in the selected regions.

Zoom Menu

- *Zoom…* – causes the view to be zoomed. If only one internal rectangle is currently defined then it is used as the zoom rectangle. If more than one internal rectangle exists then a dialog box is presented to the user, listing all internal rectangles currently defined. The user can select a zoom rectangle from the list.
- *Unzoom…*– this undoes any zoom operation by redisplaying the entire video frame area.

Two popup menus are provided with this view. One of these is discussed under section 6.2.2.2 on regions. When the user right clicks the mouse button within a region this menu is displayed. A second popup menu is displayed if the user right clicks anywhere on the display that is not within a region. This popup menu contains items that replicate those in the main menu bar and is provided simply as an alternative means of accessing this functionality.

Popup Menu

- *Set layout…* – see View menu.
- *Go to next time* – see Animation menu.
- *Go to previous time* – see Animation menu.

- *Play* – see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show particles* – see View menu.
- *Show particle numbers* – see View menu.
- *Show PTV matches* – see View menu.
- *Show true matches* –see View menu.
- *Show good matches* – see View menu.
- *Show bad matches* – see View menu.
- *Show missed matches* – see View menu.
- *Show false starts* – see View menu.
- Show particle paths – see View menu.
- Show triangulation – see View menu.
- Show nearest neighbours – see View menu.
- Show info – see View menu.
- Show regions – see View menu.
- *Zoom* – see Zoom menu.
- *Unzoom* – see Zoom menu.

6.2.2.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all particle views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Line width* – the line width for the view is specified here. The default is 1.5.

- *Font size* – sets the font size. The default is 14.
- *Rotate y label* – when selected the y axis label is rotated so that the text is vertical.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Points colour* – the colour of point particles. It can be changed by clicking on the Select… button.
- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *PTV matches colour* – the colour of lines indicating PTV matches. It can be changed by clicking on the Select… button.
- *True matches colour* – the colour of lines indicating true matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *Good matches colour* – the colour of lines indicating good matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *Bad matches colour* – the colour of lines indicating bad matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *Missed matches colour* – the colour of lines indicating missed matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *False starts colour* – the colour of lines indicating false starts for particle records with true matches. It can be changed by clicking on the Select… button.
- *Particle path colour* – the colour of lines indicating a particular particle path. It can be changed by clicking on the Select… button.
- *Triangulation colour* – the colour of lines indicating the triangulation used for velocity interpolation. It can be changed by clicking on the Select… button.

- *Background colour* – the background colour. It can be changed by clicking on the Select… button.

6.2.2.2 Regions

Regions (see chapter 47) are a powerful tool for selecting areas within a particle frame. Regions are created within the particle 2D record particle view by using a range of drawing tools selectable within the Region menu of the view. Four different region geometries are available – rectangle, circle, polygon and freehand drawing. See chapter 47 for details on how regions are drawn and manipulated.

Region manipulation within a view requires the view to have the mouse and keyboard focus. A small rectangular indicator in the bottom right hand corner of the view is green when the area has the focus and red when it does not. To regain the focus simply click the mouse in the particle frame display area.

Within the particle record particle view regions have three primary uses.

Firstly, it is commonplace to want to manipulate the particles or matches within a certain region of the flow. For example, occasionally erroneous particles that are created during the particle identification process need to be removed from the record. Sometimes a small number of poor matches, remaining after a PTV analysis has been executed, also need to be removed.

Secondly, the user may wish to obtain more detailed information about the particle, matches, costs and matching statistics within a region of the flow.

Thirdly, the user may wish to zoom in on a particular rectangular region within the flow.

These actions are available by right clicking the mouse within the region of interest. Note that if the mouse lies within the area defined by more than one region, then the most recently defined region is used. Right clicking the mouse displays a popup menu with the following options.

Popup menu

- *Delete particles in region*… – deletes all particles displayed within the region. The region can be an external or internal region. In the first case all particles outside of the drawn figure will be deleted and in the second case all particles inside the figure are deleted. The particles removed

from the record are only those actually displayed in the view. Thus if only 5 frames are displayed the particles in those frames that lie within the defined region will be deleted. If all particles in a particular region are to be deleted then all frames must be displayed. It is worth noting that freehand regions can be impractical for the deletion of particles from a large number of frames containing many (1000s of) particles. The computation requirements for determining whether a particle lies within a freehand region can be very high when applied to many particles. Of course the deletion of particles from a record will automatically delete any matches involving these particles. The user is asked to confirm this action before it takes effect.

- *Delete selected particles in region…* – this option provides a more refined mechanism for deleting particles in a region. A dialog box is displayed that enables the user to select ranges of certain parameters that will determine which particles to delete. The parameters in question are the particle number, particle radius, particle colour (actually all three colour guns can be set separately), the particle aspect ratio, orientation and eccentricity. Thus if the user wished to delete all particles smaller than a certain radius the maximum radius parameter would be specified. If a parameter has no limits it takes a value of -1.
- *Delete matches in region…* – deletes all matches contained within the region. This requires both of the matched particles to lie within the region.
- *Delete unmatched particles in region…* – deletes all particles contained within the region that are not matched in a previous or next frame. This action is applied to all frames currently displayed.
- *Delete matched particles in region…* – deletes all particles contained within the region that are matched in a previous or next frame. This action is applied to all frames currently displayed.
- *Delete future paths of particles in region…* – this option deletes all of the matches in future frames comprising the particle paths originating from the particles displayed within the region. This includes particles in all of the frames currently visible in the view.
- *Delete past paths of particles in region…* – deletes all of the matches in past frames comprising the particle paths originating from the particles displayed within the region. This includes particles in all of the frames currently visible in the view.
- *Open match details view…* – displays the match details view (see section 6.2.7) based on the particles and frames displayed within the region. The match details view uses the results from a PTV 2D analysis pipeline defined for the record to compute costs. If more than one pipeline is defined for the record a dialog box prompting the user to select one of the defined pipelines will appear first.
- *Open particle details view…* – displays the particle details view (see section 6.2.6) based on the particles and frames displayed within the region.
- *Zoom…* – if the mouse is within an internal rectangle this region is used as a zoom rectangle. This means that this region defines the displayed area. It is effectively a shorthand way of changing the text fields in the view toolbar. Complications arise if the mouse lies within more than one rectangular region, or within a region which is not an internal rectangle. In the first case the most recently defined rectangle is used as the

zoom rectangle, while in the second case a dialog box is presented to the user, listing all internal rectangles currently defined. The user can select a zoom rectangle from the list.

- *Unzoom…* – this undoes any zoom operation by redisplaying the entire particle frame.

6.2.3 Process view

Description

The process view for a particle record is exactly the same as that for an image sequence. See section 3.2.3.

6.2.4 Match summary view

Description

The match summary view provides detailed information about the matches and particle paths generated by any matching analyses that have been performed on the record.

Under the Matches tab a table listing all frames and their matching statistics is displayed. For a record with true matches this will include the number (and percentage) of good, bad, missed and false matches, together with the number of particles, the number of true matches and the total number of PTV matches. At the bottom of the table averages over all frames are given. For a record without true matches only the number of PTV matches is given.

Under the Paths tab a breakdown of the length of particle paths in the record is available. The path length equals the number of particles within it. Thus a particle that is unmatched has a path length of 1. The average path length is displayed at the bottom of the table.

Access

This view is accessed through the popup menu of a record. Particle records are displayed in the *Streams* object and particle record list views.

6.2.5 Comparison view

Description

The comparison view is intended to provide the user with a mechanism by which they can compare the particles in two different particle 2D records. The most common example of the use of this view is when a

user is experimenting with particle identification algorithms. Different settings will generate different particles and this view enables the user to see how many particles the two views have in common and how many are substantially different. Two particles, one in each of the two records, are said to be the same (i.e. in common) when their locations lie within a specified tolerance of each other. This tolerance can be specified by the user. If multiple particles in the comparison record lie within the specified tolerance of a particle in the original record then the particle closest to it is chosen to be the matching particle.

The view itself is very similar to the particle view. The key difference is that particle matching information is not displayed. Instead the focus is on the particles themselves and the commonality between two different records.

The user selects the comparison record from those already loaded into *Streams* and this is done through the Comparison management view – see section 6.2.5.2.

The view has four toolbars displayed above the particle display area. The toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls that allow the user to control the animation features of the view. The buttons provided in this toolbar are common to all views with animation within the *Streams* environment.

- *Current time* – this combo box displays the first frame currently displayed as well as its time. The user can select any value in this box to move to a different time in the field.
- *Number of frames displayed* – this combo box selects the number of frames (and corresponding time) overlaid in the view.
- *Go to next time* – steps the display forward one frame.
- *Go to previous time*– steps the display back one frame.
- *Play*– starts the animation forward in time.
- *Play in reverse* – starts the animation backward in time.
- *Pause* – pauses the animation.
- *Go to start* – resets the display to the first frame.
- *Go to end* – resets the display to the last frame.
- *Animation timing delay (ms)* – this text box allows the user to change the speed of the animation. The displayed timing delay is the time step for

the record. The actual delay may be longer if the drawing overhead is high.

Particle toolbar

The particle tool bar sits below the animation toolbar. It allows the user to select the particles that are displayed in the view.

- *All particles* – when this button is pressed all the particles, in both the record itself and the comparison record, in the visible frames are displayed.
- *Record particles* – when this button is pressed all the particles in the record that do not correspond to any particles in the comparison record are displayed.
- *Comparison particles* – when this button is pressed all particles in the comparison record that do not correspond to any particles in the original record are displayed.
- *Common particles* – when pressed all particles that are common to both records are displayed.

Cursor toolbar

The cursor toolbar, second from the bottom, displays the location of the cursor on screen and indicates the current drawing tool.

- *x* – This text box displays the x coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *y* – This text box displays the y coordinate (in mm) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *Drawing tool* – This text box displays the name of the currently selected drawing tool. This tool is used for defining regions in the view (see below and chapter 44). The tool can be changed through the *Region* menu.

View toolbar

The view toolbar sits at the bottom of the toolbar area. Here the user can select the portion of the frame that they wish to display. The zoom capability, described below, provides an alternative technique for changing the region of the frame that is displayed.

- *Min x axis* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x and y directions have the same length on screen.

- Max x axis – This text box displays the maximum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Min y axis – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Max y axis – This text box displays the maximum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.

Regions

This view provides the much the same support for regions as the particle view described in section 9.2.2. The only difference between the two is that there is no separate popup menu for two-dimensional rectangles drawn on the screen and hence the user does not have access to match details or particle list views.

Access

This view is accessed through the popup menu of a particle 2D record.

Menus

File Menu

- *Save as image…* – allows the user to save the particle view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user can choose the image size (in pixels) and and is able to preview the image before it is stored on disk. Further details of this process are provided in the User's Guide: Field Objects.
- *Save as image sequence…* allows the user to save each of the frames in the particle record as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the User's Guide: Field Objects.

View Menu

- *Set layout…* – this displays a dialog box that enables the user to set a range of display parameters common to all particle 2D record comparison views (i.e. changing these settings for this view changes them

for all particle views). Details regarding this dialog box are provided in section 6.2.5.1).

- *Draw point particles…* – a check menu item that determines whether the particles are displayed as points.
- *Draw true particles…* – a check menu item that determines whether the particles are displayed as three-dimensional spheres.
- *Show all particles* – when selected all particles, in both records, are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show record particles* – when selected all record particles with no corresponding particles in the comparison record are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show comparison particles* – when selected all comparison record particles with no corresponding particles in the record itself are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show common particles* – when selected all particles common to both records are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show frame – when selected the frame drawn around the domain containing the particles is displayed.
- Show scales – when selected the scales on the axes are displayed. These scales are challenging to draw in three-dimensional space as they overlap one another, the axes themselves and the particles depending on the camera location and orientation. Hiding them may help to make the display easier to interpret.
- Show info – this item displays the temporal information that is located at the bottom of the view.
- Show regions – when selected the currently defined 2D regions are displayed.
- *Redraw during resize* – if set, the view redraws during a resize operation. If the drawing operation is extensive, for example many frames with large numbers of particles, this might cause a lack of responsiveness. If not set redrawing does not occur and the user needs to perform some manual operation after the resize in order to force a redraw.
- Change variable labels… – this item opens a dialog box in which the labels for the x, and y variables can be set. Selecting and right clicking the text in the text boxes enables the user to select Greek or Latin characters.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y and t variables can be set. Sensible minimum values are set for these.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar.
- Adjust view size – this resizes the view window to minimize waste space. The same effect can be achieved by double clicking in the blank space on either side of the toolbars (assuming that they are visible).

Animation Menu

- *Go to next time* – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to previous time* – steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Play* – starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Comparison Menu

- *Manage comparison…* – this option displays the comparison manager view - see section 6.2.5.2 for details.

Region Menu

- *Manage regions…* – displays region 2D manager view (see chapter 47).
- *Load regions…*– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn* or a *.rgn2* extension.
- *Save regions as…*– saves all of the currently defined regions in a *.rgn2* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Draw internal rectangle* – sets the current drawing tool to an internal rectangle. This is the default.
- *Draw internal circle* – sets the current drawing tool to an internal circle.
- *Draw internal polygon* – sets the current drawing tool to an internal polygon.

- *Draw internal freehand* – sets the current drawing tool to an internal freehand.
- *Draw external rectangle* – sets the current drawing tool to an external rectangle.
- *Draw external circle* – sets the current drawing tool to an external circle.
- *Draw external polygon* – sets the current drawing tool to an external polygon.
- *Draw external freehand* – sets the current drawing tool to an external freehand.

Zoom Menu

- *Zoom* – causes the view to be zoomed using the currently displayed zoom rectangle.
- *Unzoom* – this undoes any zoom operation by redisplaying the entire video frame area.
- *Clear zoom rectangle*– this undoes any zoom operation by redisplaying the entire video frame area.

One popup menu is provided with this view. It contains items that replicate those in the main menu bar and is provided simply as an alternative means of accessing this functionality.

Popup Menu

- *Set layout…* – see View menu.
- *Go to next time* – see Animation menu.
- *Go to previous time* – see Animation menu.
- *Play* – see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show all particles* – see View menu.
- *Show record particles* – see View menu.
- *Show comparison particles* – see View menu.
- *Show common particles* –see View menu.
- Show frame – see View menu.
- Show scales – see View menu.
- Show info – see View menu.
- Show regions – see View menu.

6.2.5.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all comparison views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Line width* – the line width for the view is specified here. The default is 1.5.
- *Font size* – sets the font size. The default is 14.
- *Rotate y label* – when selected the y axis label is rotated so that the text is vertical.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *Particle colour* – the colour used for drawing all of the particles in the two records. It can be changed by clicking on the Select… button.

- *Record particle colour* – the used for drawing the original record only particles. It can be changed by clicking on the Select… button.
- *Comparison particle colour* – the colour used to draw the comparison record only particles . It can be changed by clicking on the Select… button.
- *Common particle colour* – the colour used to draw the common particles. It can be changed by clicking on the Select… button.
- *Background colour* – the background colour. It can be changed by clicking on the Select… button.
- *Use colour distance mapping* – when selected particles closer to the cameras are drawn brighter than those further away. This is only applicable when the particles are drawn as points.

6.2.5.2 Comparison management view

This view provides the user with a detailed breakdown of the particles in the record and the comparison record. The view includes 4 tabs – Statistics, Record, Comparison and Common.

Statistics

- *Particle record for comparison* – this drop down box lists all of the particle 2D records currently open in *Streams* that have the same number of frames. The user selects the record from this list that they wish to compare to the original record.
- *Tolerance* – this text box is used to enter the tolerance that determines whether two particles in the two records correspond to the same particle. Each particle in the original record is compared to all particles in the comparison record. The particle that lies closest to the original particles, and that is closer than the specified tolerance is designated as the same particle and added to the list of common particles. If no such particle exists then the particle is added to the list of record particles (that have no equivalent in the comparison record).
- *Number of record particles* – this is the number of particles (in all frames currently displayed) in the original record for which there are no corresponding particles in the comparison record.
- *Number of comparison particles* – this is the number of particles (in all frames currently displayed) in the comparison record for which there are no corresponding particles in the original record.
- *Number of common particles* – this is the number of particles (in all frames currently displayed) common to both records.
- *Compute statistics* – when this button is pressed the particle statistics are computed and updated in the view.

Record

- *Record particle list* – this list box displays all of the record particles (without corresponding particles in the comparison record) together with their location, size and colour.

Comparison

- *Comparison particle list* – this list box displays all of the comparison record particles (without corresponding particles in the original record) together with their location, size and colour.

Common

- *Common particle list* – this list box displays all of the common particles together with their location, size and colour.

Menus

View Menu

- Change decimal places… – this item opens a dialog box in which the number of decimal places used to display the data can be changed.

6.2.6 Diagnostic view

Description

The diagnostic view is intended to provide the user with a mechanism by which they can understand the impact of adopting different particle matching strategies on the matches in a particle record. Essentially the view computes comparison statistics between the matches for this record (termed the original record) and those for another record (termed the comparison record) that is assumed to contain exactly the same set of particles.

The view provides 4 tabs. These are described below.

Selection

This tab allows the user to select the comparison record with which the original record will be compared. In addition it allows the user to set certain analysis parameters.

- *Particle record for comparison* – this dropdown box lists all of the particle records currently open in *Streams* that possess the same number of

frames as this particle record. Each of these frames must possess exactly the same number of particles as this particle record's frames. The user selects the comparison record by choosing a record from this box. Typically the user will create a number of copies of the particle record of interest and analyse each of the records with a different set of PTV strategies. These records can be compared by ensuring that they are all open in *Streams* concurrently.

- *Number of rows* – the distributions tab enables the user to view spatial distributions of some of the comparison statistics. To display these distributions particle statistics need to be bundled together into cells (small rectangle regions). A rectangular grid of cells is used and this text field allows the user to set the number of rows of cells used.
- *Number of columns* – the distributions tab enables the user to view spatial distributions of some of the comparison statistics. To display these distributions particle statistics need to be bundled together into cells (small rectangle regions). A rectangular grid of cells is used and this text field allows the user to set the number of columns of cells used.
- *Compute diagnostic statistics* – the diagnostic statistics are computed once this button is pressed.

Matches

This tab provides the diagnostic statistics based on the matches in each frame. The data are presented in a table with the following columns.

- *Frame* – the number of the frame (starting at 0).
- *Particles* – the number of particles in the frame (must be the same for both records).
- *# of orig matches* – the number of the particles in the frame that are matched in the original record. This is also given as a percentage.
- *# of comp matches* – the number of particles in the frame that are matched in the comparison record. Also given as a percentage.
- *# of common matches* – the number of particles in the frame that are matched to the same particle in both of the particle records. Also given as a percentage.
- *# of conflict matches* – the number of particles in the frame that are matched to different particles in the two records. Also given as a percentage.
- *# of orig only matches* – the number of particles in the frame that are matched in the original record but not matched in the comparison record. Also given as a percentage.
- *# of comp only matches* – the number of particles in the frame that are matched in the comparison record but not matched in the original record. Also given as a percentage.

Velocities

This tab provides diagnostic statistics based on the particle velocities computed from the matches in each frame. In this view the symbol |du|

refers to the magnitude of the difference in velocity between the predicted particle velocities based on the particle matches in each of the records. If the two particle records have the same matches for a particle then this value will be 0.The data are presented in a table with the following columns.

- *Frame* – the number of the frame (starting at 0).
- *Particles* – the number of particles in the frame (must be the same for both records).
- *Number of conflicts* – the number of the particles in the frame that are matched differently in the two records.
- Ave |du| all (unit) – |du| averaged over all particles in the frame. This is also given as a percentage of the particle velocity.
- Ave |du| conflict (unit) – |du| averaged over all particles that have conflicting matches in the frame. This is also given as a percentage of the particle velocity.
- Std |du| conflict (unit) – the standard deviation of |du| for all particles with conflicting matches.
- Max |du| conflict (unit) – the maximum value of |du| for the particles with conflicting matches.
- Min |du| conflict (unit) – the maximum value of |du| for the particles with conflicting matches.
- Num |du| conflict > 0 – the number of particles with conflicting matches for which the original particle record predicts a higher magnitude of the particle velocity.
- Num |du| conflict <= 0 – the number of particles with conflicting matches for which the original particle record predicts a lower or equal magnitude of the particle velocity.

Distributions

The distributions tab provides spatial information about the comparison statistics in contrast to the temporal statistics provided in the Matches and Velocities tabs. Often it is an understanding of the regions within the flow where particle matches differ that is important when trying to select the best matching strategies.

In this tab a graph, that closely resembles a scalar 2D field contour view, presents time-averaged statistics for each of the cells in the grid defined in the Selection tab. The user can choose to display a number of different statistics through this graph – these are: number of conflicts, Ave |du| all, Ave |du| conflict and Max |du| conflict and these all have the same meaning as was provided in the previous tabs.

The user can change the scale used for the colour legend used in the plot by adjusting the Min value and Max value text fields. The grid can be displayed by selecting the grid button.

Access

This view is accessed through the popup menu of a record. Particle records are displayed in the *Streams* object and particle record list views.

6.2.7 Particle details view

Description

The particle details view is designed to give the user access to particle information. The particle details view displays, in tabular form, details of the particles that are contained in a region. Details for each frame displayed in the record's particle view (up to a maximum of 10) are given on different tabs. The information displayed includes the particle state information if it is available. Right clicking any particle in the list allows the user to display particle views (see chapter 4).

Access

This view is accessed through the popup menu for the region in the particle view.

6.2.8 Match details view

Description

The match details view, and its two associated views, the candidate matches view and particle path view, are designed to give the user access to internal details of any PTV matching analyses that have been performed. The match details view displays, in tabular form, details of the particles that are contained in a region. Details for each frame displayed in the record's particle view (up to a maximum of 10) are given on different tabs. The information displayed includes the state information for each particle (number, location, radius and colour (RGB)), the particles to which each particle is matched in the previous and next frames, together with their state information, and the costs associated with these matches. These costs are calculated using a PTV pipeline defined for this particle record. If more than one pipeline is defined the user selects the desired one when the view is opened.

At the top of each tab a panel displays some basic statistics associated with the particles in the frame.

Access

This view is accessed through the popup menu for a region within the particle view.

Menus

File Menu

- *Save to text file* – allows the user to save the tabulated data in this view to a text file (.txt or .csv).

View Menu

- *Open candidate matches* view – displays a candidates match view for the particle selected in the match details view.
- *Open particle path* view – displays a particle path view for the particle selected in the match details view.

6.2.9 Candidate matches view

Description

The candidate match view for a particle, p1 say, provides the user with detailed information about the potential matches for this particle in both the previous and next frames. All of the particles that were considered as potential matches to p1 in the next and previous frames are tabulated, together with their numbers, positions and the numbers of the particles to which each is currently matched. Candidates are determined purely by whether they lie within the search rectangle for the PTV 2D analysis. The particle that is actually matched to p1, if it exists, is highlighted in the table.

Also tabulated are the various costings used in the PTV 2D analysis, and their contributions to the total cost. If the total cost is above the maximum matching cost (MMC) this is indicated in the table.

At the top of the view all PTV 2D analyses defined for the selected PTV 2D analysis pipeline are listed and the analysis that has been used to calculate the costs for each potential match is highlighted. By selecting another analysis the user can view the costs associated with this analysis.

This feature provides the user with a powerful tool for exploring the possible costs that would eventuate if different analyses were used. However, it must be recognized that when a matching based costing is used in calculating the cost the current matches are used in evaluating the cost, not the matches that may have existed when the analysis actually executed.

A number of statistics associated with the potential matches are displayed at the top of each tab.

Access

This view is accessed through the View menu of the match details view.

Menus

File Menu

- *Save to text file* – this saves the view information to a comma (,) delimited text file of the user's choice.

6.2.10 Particle path view

Description

The particle path view provides a tabulated list of the particles linked to the chosen particle through matches in previous and subsequent frames. Particle paths are like chains, with each particle in the chain linked to another in the next or previous frame. This view will look much the same for any particle in a particular path. Details about each particle in the path are also listed.

Access

This view is accessed through the View menu of the match details view.

Menus

File Menu

- *Save to text file* – This saves the content of the view to a comma (,) delimited text file of the user's choice.

6.3 Popup menu

The particle 2D record provides the following options in its popup menu.

- *Save as…* – saves the particle record in a new file. All particle record files have the extension *.par2*. The new file name is displayed next to the particle record name in the *Streams* object list view.
- *Save* – saves the particle record to disk. If the particle record currently has no associated file then the user is prompted for a file name. All particle record files have the extension .par2.
- *Save Eulerian velocities to text file…* – saves the velocities of each particle in each frame to a comma (,) delimited text file (*.txt* or *.csv*). The user is at liberty to choose the frames that are to be included as well as whether the velocity can be calculated from a single match in the previous or next frame, or whether it requires matches in both frames. The data is output by frame.
- *Save Lagrangian velocities to text file…* – saves the velocities of each particle as it moves to a comma (,) delimited text file (*.txt* or *.csv*). Unlike the Eulerian data the Lagrangian data produces velocities along each particle path, instead of the velocities at each time step. By definition all velocities will be calculated from two matches, except for the first and last velocities in the path. These will have only a single match available. The data is output by particle path not by frame.
- *Create copy* – creates a copy of this particle record and adds it to the *Streams* system. The copied record includes all particle data but does not include the processes of the original particle record.
- *Change name…* – allows the user to change the name of the particle record. This name change will be reflected in the *Streams* object list view. This action can also be performed through the details view.
- *Edit description…* – allows the user to change the description of the particle record. This action can also be performed through the details view.
- *Delete frames…* – this option allows the user to delete frames from a particle record. The first and last frames to be deleted are selectable by the user. The deletion of frames will remove affected particle matches. The frames are renumbered after the deletion if appropriate. This action is drastic and the user is prompted for confirmation.
- *Edit frames…* – this option allows the user to thin the number of frames by retaining only 1 frame in every N where the user can specify the value of N. Most importantly if a particle in frame I, that is to be retained, has a particle path that extends to frame i+N then a match for this particle between frames I and i+N is created based on this path. Thus deleing frames in this way does not necessarily remove all matches. Clearly the larger the value of N the more likely that particle tracks will be lost. This action is drastic and the user is prompted for confirmation.
- *Compress domain…* – this option allows the user to compress the physical domain of the particle record. When a record is transformed extensions to its physical domain may be made. If the result of these transformations is such that there are regions in the particle record that

contain no particles then the compress option will reduce the physical domain of the record to the smallest rectangle that contains all particles. The user is prompted as to whether they wish to re-zero the bottom left corner of the domain so that it corresponds to the origin.

- *Segment frames…* – this option allows the user to create a segmentation for the particle record. The user is prompted to specify the number of rows and columns in the segment array. Once the segmentation is created this menu option is ticked and selecting it again will remove the segmentation, thus freeing up the memory resources consumed by the segmentation.
- *Sort particles…* – this option allows the user to sort the particles within a frame. There are four submenus that specify how the particles are to be sorted. The particles can be order by increasing x, decreasing x, increasing y or decreasing y. This only changes the particle numbering, no matches or other particle properties are changed.
- *Delete matches…* – this option deletes all particle matches. The user is prompted for confirmation of the choice.
- *Delete all matched particles…* – this option allows the user to delete all particles within the record that are currently matched. The user is prompted for confirmation.
- *Delete all unmatched particles…* – this option allows the user to delete all particles within the record that are currently unmatched. The user is prompted for confirmation.
- *Delete random particles…* – this option allows the user to delete a random selection of particles in the record. The user is prompted to specify the proportion of particles to delete (0-1). The process starts at frame 0 deleting the proportion of particles requested from that frame. If these particles are matched in subsequent frames the matched particles are deleted from those frames as well. Thus the number of new particles deleted in subsequent frames will be reduced based on the number of particles that have already been removed through matches to particles in previous frames. Note that as the particles selected to be deleted are chosen based on random numbers the exact number of particles removed is unlikely to match the requested proportion exactly. The user is prompted for confirmation of this action.
- *Create true matches* – (Only available if the particle record has not been created from a theoretical model). This option allows the user to artificially create a set of true matches for particles in the particle record. The current matches are taken to be the true matches. The creation of true matches will result in the particle record requiring more memory, both internal and on disk, for its storage.
- *Remove true matches* – (Only available if the particle record has not been created from a theoretical model). This option allows the true matches created with the *Create true matches* option to be deleted.
- *Open details view* – opens the details view.
- *Open particle view* – opens the particle view. This is the default view.
- *Open process view* – opens the process view.
- *Open match summary view* – opens the match summary view.
- *Open comparison view* – opens the comparison view.

- *Open diagnostic view* – opens the diagnostic view.
- *Execute process pipeline* – executes the process pipeline if it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the record from *Streams*. All views are closed. If the record has unsaved data then the user is prompted to save this data before closure.

7 Particles in 3D

7.1 Description

A **particle 3D** object represents a particle identified from a particle 2D record bundle based on images from a number of cameras, or created artificially by a theoretical model. Each particle is assigned a unique number (starting at 0) in the particle frame in which it appears. The following data is always stored with the particle:

- *x, y, z* – the position of the particle. The unit for these and other spatial quantities is dependent on the spatial unit specified in the associated particle 3D record (see chapter 9).
- *radius* – the equivalent radius of the particle. This is computed from the calculated areas of the particle in the original images assuming the particle is spherical.
- *colour* – red, green and blue intensities of the particle each in the range 0-255 even if the original image has higher resolution. All three intensities are the same for a monochrome image.

Particles are used in particle tracking analysis. Thus they also store the particles to which they are matched in the previous and next frames. Until a PTV 3D analysis creates these matches they are undefined. Particles that are created from theoretical models also know which particles in the previous and next frames are their correct, or true, matches. It is possible for the user to artificially create true matches for particles created from digital images (see chapter 9) using the current set of particle matches. In this case such particles are treated as if they were derived from a theoretical model.

7.2 Views

A particle object has two views.

- *Details view.*
- *Path view.*

7.2.1 Details view

Description

The details view displays the data, as described above, pertaining to a particle. These details cannot be edited by the user.

Access

It is accessed through

- the particle list view of a particle 3D record.
- the particle matches view of a particle 3D frame.

7.2.2 Path view

Description

The path view displays information about a particle path throughout a particle record. A particle path is defined as the set of particles across a number of frames that are linked by particle matches. If such a path has a gap because no match exists in the subsequent or previous frame, the path stops at that frame. This view presents a table of all the particles in the path of the particle, showing the particle information, location, colour etc. for each particle in the path. If the particle has true matches then the true path is listed alongside that produced by the particle matching process.

Access

It is accessed through

- the particle list view of a particle 3D record.
- the particle matches view of a particle 3D frame.

8 Particle Frames in 3D

8.1 Description

A **particle 3D frame** object represents a frame of particles in three-dimensional space. It contains a list of particles identified in the creation of a particle 3D record from a particle 2D record bundle, or artificially produced by a theoretical model.

Each particle 3D frame is part of a particle 3D record and is assigned a unique number (starting at 0) within that record.

8.2 Views

The particle 3D frame provides four views. These are

- details view
- particle list view
- matches view
- segmentation view

8.2.1 Details view

Description

This view displays details of the particle frame, including the number of particles within it. The details displayed vary depending on whether the particles within the frame have true matches recorded or not (i.e. have been created by a theoretical model or have had true matches created artificially). For a frame with true matches, details about the number of correct and incorrect matches are included.

Access

This view is accessed through the popup menu of a particle 3D frame. Frames for a record are displayed in a particle 3D record details view.

8.2.2 Particle list view

Description

This view displays a table of the particles within the frame. For each particle all of the available particle information is tabulated – position, radius, and colour (RGB).

Access

This view is accessed through the popup menu of a particle 3D frame. Frames for a record are displayed in a particle 3D record details view.

Menus

Sort Menu

- *By number* – sorts the table by particle number.
- *By x* – sorts the table by x.
- *By y* – sorts the table by y.
- *By z* – sorts the table by z.
- *By radius* – sorts the table by particle radius.
- *By intensity* – sorts the table by particle intensity.

8.2.3 Matches view

Description

The matches view for a particle 3D frame comprises a table giving particle matching information. The particles in the frame are listed in numerical order and their details (location, radius and colour) are included with them. In the second and third columns the particles that these particles are matched to in the previous and next frames are displayed along with their positions. If the particles in the frame have true match information then the particle numbers of the true matches are also included in these columns. In the final column the path length of each particle is listed. A path length of 1 indicates that the particle is not matched to particles in either the previous or next frames.

This is the default view for a particle 3D frame.

Access

This view is accessed through the popup menu of a particle 3D frame or by double clicking the frame. Frames for a particle 3D record are displayed in a particle 3D record details view.

Menus

Particle Menu

- *Display particle details view…* – displays the details view of a selected particle in the table.
- *Display particle path view…* – displays the path view of a selected particle in the table.

8.2.4 Segmentation view

Description

Many of the processes involving particle 3D records, and hence particle 3D frames, are expedited if particles can be rapidly located in the frame. **Segmentation** of particle frames is a mechanism for doing this. A **segment** is a three dimensional rectangular region within a particle frame that records the particles that lie within the region. The segmentation of a particle frame comprises a rectangular mesh of segments that cover the entire frame. A segmentation is created through the popup menu of the particle 3D record and at that time the number cells in x, y and z directions can be defined.

The segmentation of a particle frame will provide substantial speed improvements in a range of particle tracking velocimetry processes – for example finding candidate particles for matching – but it does carry with it an extra storage overhead, typically 20%.

The segmentation view provides a list of the segments in the frame's segmentation and for each segment it lists the particles in that segment.

Access

This view is accessed through the popup menu of a particle 3D frame. Frames for a particle 3D record are displayed in a particle 3D record details view.

8.3 Popup menu

The popup menu is displayed by right clicking on the particle 3D frame object when it is displayed in the particle 3D record details view. The menu options are

- *Save particle details to text file…* – saves all particle information to a comma (,) delimited text file selected by the user.
- *Open details view* – opens the details view of the frame.
- *Open particle list view* – opens the particle list view of the frame.
- *Open matches view* – opens the matches view of the frame.
- *Open segmentation view* – opens the segmentation view of the frame.

9 Particle Records in 3D

9.1 Description

The particle 3D record is the key object when dealing with three-dimensional particle tracking. A particle 3D record represents a sequence of frames containing three-dimensional particles obtained, either from a particle 2D record bundle, from a theoretical model, or created from a text file. It contains a list of particle 3D frames (see chapter 8) each of which, in turn, contains the particles that have been identified in that frame. The record needs to store a number of basic experimental parameters such as the time step between frames and the size of the spatial domain that contains all of the particles within the record.

Each particle record is allocated a name by the user and a textual description can also be specified by the user.

The segmentation of particle records, as discussed in section 8.2.4, can lead to substantial reductions in computation time for certain PTV operations. While segmentation carries a modest storage overhead, and a small computational expense (typically a few seconds to create the segmentation), the execution speed improvements can be very significant – more than an order of magnitude in some cases – and therefore the segmenting of particle records should be adopted as standard practice if no memory issues exist. Particle records are segmented by default.

9.2 Views

The particle record provides eight different views of the data it stores. These are:

- details view
- particle view
- process view
- match summary view
- comparison view
- diagnostic view
- particle list view
- match details view

9.2.1 Details View

Description

The details view for a particle record displays key information about the record, including its size, spatial domain, time step, segmentation etc.

The details view allows the user to change a number of the record parameters. This includes the name of the record, the description, the temporal and spatial units and the time step between frames. The user must press the Save button for these changes to take effect. If the changes are to be discarded the Cancel button is pressed.

The view has four tabs, Parameters, Description, Frames and Segmentation. These are described below.

Parameters

- *Record name* – the name associated with the record. This can be changed in the details view or through the object's popup menu. Note that this is not the name of the file in which the record is stored, although of course the user could choose to make them the same.
- *Record type* – this field is not editable. It specifies whether the record was generated from a model or from images.
- *True matches exist* – this box is ticked when true matches exist for this record. It cannot be edited.
- *Temporal unit* – the user defined time unit. The default is s (seconds).
- *Time step (unit)* – the time step between frames in the units defined above.
- *Number of frames* – the user number of frames in the record. This is not editable by the user.
- *Spatial unit* – the user defined space unit. The default is mm (millimetres).
- *x min (unit)* – the minimum value of x for the record. All particles have an x coordinate equal to or larger then this. This is not editable.
- *x max (unit)* – the maximum value of x for the record. All particles have an x coordinate equal to or less then this. This is not editable.
- *y min (unit)* – the minimum value of y for the record. All particles have an y coordinate equal to or larger then this. This is not editable.
- *y max (unit)* – the maximum value of y for the record. All particles have an y coordinate equal to or less then this. This is not editable.
- *z min (unit)* – the minimum value of z for the record. All particles have an z coordinate equal to or larger then this. This is not editable.
- *z max (unit)* – the maximum value of z for the record. All particles have an z coordinate equal to or less then this. This is not editable.

Description

- *Description* – the user can provide a description of the record in the text box provided.

Frames

- *Frames* – a list of the particle 3D record frames contained in the record is provided. The user can access the popup menus of the frames in this view and hence get access to particle information through the frame views.

Segmentation

- *Segments are used* – this box is ticked if segmentation of the record has occurred. This cannot be changed in the view but can be changed through the popup menu of the record.
- *Number of x segments* – the number of segments used to divide the x axis. This is not editable.
- *Number of y segments* – the number of segments used to divide the y axis. This is not editable.
- *Number of z segments* – the number of segments used to divide the z axis. This is not editable.

Access

This view is accessed through the popup menu of a particle 3D record. Particle 3D records are displayed in the *Streams* object and particle record list views.

9.2.2 Particle view

Description

The particle view is the most useful of the particle record views, and it gives rise to two more views – the particle list and the match details views. It displays an idealised three-dimensional picture of the particle record, representing particles in each frame as points on the screen. It possesses all of the key behaviours of the particle view of a particle 2D record with the ability to display multiple frames simultaneously and to view the record as a video.

Another particle 3D record can be dropped onto this view. If this operation is performed without the ALT/OPTION key pressed the particles in the dropped record will be overlaid on those in the original record – thus all particles in frame 0 of the dropped record will be added

to frame 0 of the original record and so on. No checking for compatibility of the two records is undertaken. If the ALT/OPTION key is pressed during this operation the frames of the dropped record are concatenated to the original record – thus if the original record has N frames the frames of the dropped record will become frames N+1 and beyond. The first of these operations cannot be reversed so the user should always make a copy of the original record before undertaking this action. The second action can be reversed by simply deleting the new frames. In both cases the user is asked to confirm the action before it is executed. The dropped record is unaffected by either of these operations.

As this view provides a three-dimensional view it supports the features of three-dimensional views described in chapters 1, 2 and 3 of the *User's Guide: Field Objects* manual. Please refer to this manual for details.

The view has five toolbars displayed above the particle display area. The toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls that allow the user to control the animation features of the view. The buttons provided in this toolbar are common to all views with animation within the *Streams* environment.

- *Current time* – this combo box displays the first frame currently displayed as well as its time. The user can select any value in this box to move to a different time in the field.
- *Number of frames displayed* – this combo box selects the number of frames (and corresponding time) overlaid in the view.
- *Go to next time* – steps the display forward one frame.
- *Go to previous time* – steps the display back one frame.
- *Play* – starts the animation forward in time.
- *Play in reverse* – starts the animation backward in time.
- *Pause* – pauses the animation.
- *Go to start* – resets the display to the first frame.
- *Go to end* – resets the display to the last frame.
- *Animation timing delay (ms)* – this text box allows the user to change the speed of the animation. The displayed timing delay is the time step for the record. The actual delay may be longer if the drawing overhead is high.

Particle toolbar

The particle toolbar sits below the animation toolbar. It allows the user to select the particle and particle matching information that is displayed in the view. The buttons present in this toolbar will depend on whether the particle record has true matches information.

- *Particles* – when this button is pressed all the particles in the visible frames are displayed.
- *Numbers* – when this button is pressed all the particle numbers are displayed next to the particles.
- *PTV* – when this button is pressed all matches between particles in the visible frames (generated by a PTV 3D analysis) are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *True* – (Only visible if true matches information is available). When this button is pressed the true matches between particles in the visible frames are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *Good* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames that match the true matches are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *Bad* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames for which different true matches exist (ie the PTV matches are not the correct matches) are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *Missed* – (Only visible if true matches information is available). When this button is pressed the true matches between particles in the visible frames that have been missed by a PTV analysis are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *False* – (Only visible if true matches information is available). When this button is pressed the PTV matches between particles in the visible frames that have no corresponding true match are drawn as lines between the particles. The colour can be changed in the view layout dialog box (see section 9.2.2.1).
- *Paths* – When this button is pressed the particle paths generated by a PTV analysis, that include the particles in the first visible frame, are drawn as coloured lines. The particles present in these paths do not need to be currently displayed. Each pathline is given a random colour that will change if the user moves through the frames.

x toolbar

- *x min* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- x max – This text box displays the maximum value of the x axis currently displayed.
- *Focal x* – This text box displays the x coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

y toolbar

- *y min* – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- y max – This text box displays the maximum value of the y axis currently displayed.
- *Focal y* – This text box displays the y coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

z toolbar

- *z min* – This text box displays the minimum value of the z axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- z max – This text box displays the maximum value of the z axis currently displayed.
- *Focal z* – This text box displays the z coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

The particle view is the default view for a particle 3D record.

Access

This view is accessed through the popup menu of a particle 3D record or by double clicking the object.

Menus

File Menu

- *Save as image…* – allows the user to save the particle view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user can choose the image size (in pixels) and and is able to preview the image before it is stored on disk. Further details of this process are provided in the User's Guide: Field Objects.
- *Save as image sequence…* allows the user to save each of the frames in the particle record as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the User's Guide: Field Objects.

View Menu

- *Set layout…* – this displays a dialog box that enables the user to set a range of display parameters common to all particle 3D record particle views (i.e. changing these settings for this view changes them for all particle views). Details regarding this dialog box are provided in section 9.2.2.1).
- *Draw point particles…* – a check menu item that determines whether the particles are displayed as points.
- *Draw true particles…* – a check menu item that determines whether the particles are displayed as three-dimensional spheres.
- *Draw monochrome particles* – if selected the particles are drawn in a single colour (which is set in the view layout – see section 9.2.2.1). This is only available when the particles are drawn as points.
- *Draw true coloured particles* – if selected the particles are drawn in their true colour. This is only available when the particles are drawn as points.
- *Draw colour coded particles…* – if selected the particles are drawn in a single colour except for the first and last frames displayed. Particles in these frames are coloured green and red respectively. This is only available when the particles are drawn as points
- *Use sorted rendering* – when particles are drawn as points there is no assurance that the particle closest to the camera will be drawn on top. This option sorts the particles before rendering them to ensure the correct ordering is performed. There is a computational costs associated with this. This option is not relevant when the particles are drawn as spheres as these render correctly.
- *Display subset of particles...* – a dialog box is displayed that allows the user to select the particles to display – 1 particle in every N particles is displayed and the user specifies N.

- *Show particles* – a check menu item that determines whether the particles are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show particle numbers* – if selected the particle numbers are displayed alongside the particles in the view. This tends to useful only with modest numbers of particles or when the user has zoomed in on a portion of the flow.
- *Show PTV matches* – a check menu item that determines whether the PTV matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show true matches* – (only visible if true match information is available) a check menu item that determines whether the true matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show good matches* – (only visible if true match information is available) a check menu item that determines whether the good matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show bad matches* – (only visible if true match information is available) a check menu item that determines whether the bad matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show missed matches* – (only visible if true match information is available) a check menu item that determines whether the missed matches are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show false starts …* – (only visible if true match information is available) a check menu item that determines whether the false starts are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show particle paths – a check menu item that determines whether the particle paths are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show frame – when selected the frame drawn around the domain containing the particles is displayed.
- Show scales – when selected the scales on the axes are displayed. These scales are challenging to draw in three-dimensional space as they overlap one another, the axes themselves and the particles depending on the camera location and orientation. Hiding them may help to make the display easier to interpret.
- Show triangulation – when selected the three-dimensional triangulation (comprising tetrahedral) used for velocity field calculation for the first frame is displayed in the view. Depending on the number of particles this triangulation can be extremely densely packed with the defining edges of the tetrahedrons.
- Show info – this item displays the temporal information that is located at the bottom of the view.
- Show regions – when selected the currently defined 3D regions are displayed.

- *Redraw during resize* – if set, the view redraws during a resize operation. If the drawing operation is extensive, for example many frames with large numbers of particles, this might cause a lack of responsiveness. If not set redrawing does not occur and the user needs to perform some manual operation after the resize in order to force a redraw.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y, z and t variables can be set. Sensible minimum values are set for these.
- Reset min/max to default – this resets the x, y and z axis ranges to their defaults.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar. This can also be effected by double clicking the mouse while holding down the ALT key.

Animation Menu

- *Go to next time* – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to previous time* – steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Play* – starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Cameras menu

- *Manage cameras…* – opens the camera management view. See chapter 2 of the *User's Guide: Field Objects* manual.
- *Open current camera controller* – this opens the controller view of the currently selected camera so the user can make adjustments to the camera settings. These changes are often most easily achieved using keyboard commands (see chapter 2 of the *User's Guide: Field Objects* manual). The camera controller view is also opened by double clicking the mouse on the display area with SHIFT and ALT keys held down.

Lighting menu

- *Manage lighting…* – this option opens the manage lighting view. See chapter 3 of the *User's Guide: Field Objects* manual.

Region Menu

- *Manage regions…* – displays region 3D manager view (see chapter 48) and enables the user to create 3D regions in the view.
- *Load regions…*– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn3* extension.
- *Save regions as…*– saves all of the currently defined regions in a *.rgn3* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Delete particles in selected regions* – this will delete all particles that lie within the selected 3D regions for the frames currently displayed. The user will be asked to confirm the action.

Zoom Menu

- *Zoom* – causes the view to be zoomed using the currently displayed zoom rectangle.
- *Unzoom* – this undoes any zoom operation by redisplaying the entire video frame area.
- *Clear zoom rectangle*– this undoes any zoom operation by redisplaying the entire video frame area.

Two popup menus are provided with this view. The first of these is discussed under the section on regions (9.2.2.2). When the user right clicks the mouse button within a rectangle drawn on the screen this menu is displayed. However a second popup menu is displayed if the user right clicks anywhere on the display that is not within this rectangle. This popup menu contains items that replicate those in the main menu bar and is provided simply as an alternative means of accessing this functionality.

Popup Menu

- *Set layout…* – see View menu.
- *Go to next time* – see Animation menu.
- *Go to previous time* – see Animation menu.
- *Play* – see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show particles* – see View menu.
- *Show particle numbers* – see View menu.
- *Show PTV matches* – see View menu.
- *Show true matches* –see View menu.
- *Show good matches* – see View menu.
- *Show bad matches* – see View menu.
- *Show missed matches* – see View menu.
- *Show false starts* – see View menu.
- Show particle paths – see View menu.
- Show frame – see View menu.
- Show scales – see View menu.
- Show triangulation – see View menu.
- Show info – see View menu.
- Show regions – see View menu.

9.2.2.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all particle views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Number of x divisions* – this parameter specifies how many divisions appear along the x axis and hence how many scale values will be displayed.
- *Number of y divisions* – this parameter specifies how many divisions appear along the y axis and hence how many scale values will be displayed.
- *Number of z divisions* – this parameter specifies how many divisions appear along the z axis and hence how many scale values will be displayed.
- *Radius factor* – when the particles are drawn as spheres their radii are multiplied by this factor. This gives the user some control over how large the particles appear in the view.
- *Geometric scale factor* – this option allows the user to globally change the "size" of objects such as the axes thickness.
- *Font size* – sets the font size. The default is 14.
- *Draw axes at origin* – the default setting is to draw the axes to the left, bottom and back of the domain. By selecting this checkbox the axes are relocated so they pass through the origin.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Points colour* – the colour of point particles. It can be changed by clicking on the Select… button.
- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *PTV matches colour* – the colour of lines indicating PTV matches. It can be changed by clicking on the Select… button.
- *True matches colour* – the colour of lines indicating true matches for particle records with true matches. It can be changed by clicking on the Select… button.

- *Good matches colour* – the colour of lines indicating good matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *Bad matches colour* – the colour of lines indicating bad matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *Missed matches colour* – the colour of lines indicating missed matches for particle records with true matches. It can be changed by clicking on the Select… button.
- *False starts colour* – the colour of lines indicating false starts for particle records with true matches. It can be changed by clicking on the Select… button.
- *Triangulation colour* – the colour of lines indicating the triangulation used for velocity interpolation. It can be changed by clicking on the Select… button.
- *Background colour* – the background colour. It can be changed by clicking on the Select… button.
- *Use colour distance mapping* – when selected particles closer to the cameras are drawn brighter than those further away. This is only applicable when the particles are drawn as points.

9.2.2.2 Regions

Both two-dimensional (see chapter 47) and three-dimensional regions (see chapter 48) are used in a number of ways in this view. Three-dimensional regions, which are created and managed through the Region menu are typically used for removing particles from the record. Unlike 2D regions these are not easily drawn on screen and thus must be defined by entering their parameters. Two-dimensional rectangular regions are used for zooming in and out in the view, and for selecting particles whose details are of interest. In both cases these features are enabled by drawing rectangular regions on the display in the usual way (see chapter 47). The two-dimensional rectangle is always interpreted as residing in the focal plane of the current camera.

When the user right clicks the mouse in a 2D rectangular region the following popup menu is displayed.

- *Open match details view…* – displays the match details view (see section 9.2.6) based on the particles and frames displayed within the drawn rectangle.
- *Open particle list view…* – displays the particle list view (see section 9.2.7) based on the particles and frames displayed within the drawn rectangle.
- *Zoom…* – zooms the view.
- *Unzoom…* – this undoes any zoom operation.

9.2.3 Process view

Description

The process view for a particle record is exactly the same as that for an image sequence. See section 3.2.3

9.2.4 Match summary view

Description

The match summary view provides detailed information about the matches and particle paths generated by any matching analyses that have been performed on the record.

Under the Matches tab a table listing all frames and their matching statistics is displayed. For a record with true matches this will include the number (and percentage) of good, bad, missed matches and false starts, together with the number of particles, the number of true matches and the total number of PTV matches. At the bottom of the table averages over all frames are given. For a record without true matches only the number of PTV matches is given.

Under the Paths tab a breakdown of the length of particle paths in the record is available. The path length equals the number of particles within it. Thus a particle that is unmatched has a path length of 1. The average path length is displayed at the bottom of the table.

Access

This view is accessed through the popup menu of a record. Particle 3D records are displayed in the *Streams* object and particle record list views.

9.2.5 Comparison view

Description

The comparison view is intended to provide the user with a mechanism by which they can compare the particles in two different particle 3D records. The most common example of the use of this view is when a user is experimenting with particle identification algorithms for a particle 2D record bundle. Different settings will generate different particles and this view enables the user to see how many particles the two views have in common and how many are substantially different. Two particles, one in each of the two records, are said to be same (i.e. in common) when

their locations lie within a specified tolerance of each other. This tolerance can be specified by the user.

The view itself is very similar to the particle view. The key difference is that particle matching information is not displayed. Instead the focus is on the particles themselves and the commonality between two different records.

The user selects the comparison record from those already loaded into *Streams* and this is done through the Comparison management view – see section 9.2.5.2.

The view has five toolbars displayed above the particle display area. The toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls that allow the user to control the animation features of the view. The buttons provided in this toolbar are common to all views with animation within the *Streams* environment.

- *Current time* – this combo box displays the first frame currently displayed as well as its time. The user can select any value in this box to move to a different time in the field.
- *Number of frames displayed* – this combo box selects the number of frames (and corresponding time) overlaid in the view.
- *Go to next time* – steps the display forward one frame.
- *Go to previous time*– steps the display back one frame.
- *Play*– starts the animation forward in time.
- *Play in reverse* – starts the animation backward in time.
- *Pause* – pauses the animation.
- *Go to start* – resets the display to the first frame.
- *Go to end* – resets the display to the last frame.
- *Animation timing delay (ms)* – this text box allows the user to change the speed of the animation. The displayed timing delay is the time step for the record. The actual delay may be longer if the drawing overhead is high.

Particle toolbar

The particle tool bar sits below the animation toolbar. It allows the user to select the particles that are displayed in the view.

- *All particles* – when this button is pressed all the particles, in both the record itself and the comparison record, in the visible frames are displayed.
- *Record particles* – when this button is pressed all the particles in the record that do not correspond to any particles in the comparison record are displayed.
- *Comparison particles* – when this button is pressed all particles in the comparison record that do not correspond to any particles in the original record are displayed.
- *Common particles* – when pressed all particles that are common to both records are displayed.

x toolbar

- *x min* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- x max – This text box displays the maximum value of the x axis currently displayed.
- *Focal x* – This text box displays the x coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

y toolbar

- *y min* – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- y max – This text box displays the maximum value of the y axis currently displayed.
- *Focal y* – This text box displays the y coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

z toolbar

- *z min* – This text box displays the minimum value of the z axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- z max – This text box displays the maximum value of the z axis currently displayed.

- *Focal z* – This text box displays the z coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

Regions

This view provides the much the same support for regions as the particle view described in section 9.2.2. The only difference between the two is that there is no separate popup menu for two-dimensional rectangles drawn on the screen and hence the user does not have access to match details or particle list views.

Access

This view is accessed through the popup menu of a particle 3D record.

Menus

File Menu

- *Save as image…* – allows the user to save the particle view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user can choose the image size (in pixels) and and is able to preview the image before it is stored on disk. Further details of this process are provided in the User's Guide: Field Objects.
- *Save as image sequence…* allows the user to save each of the frames in the particle record as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the User's Guide: Field Objects.

View Menu

- *Set layout…* – this displays a dialog box that enables the user to set a range of display parameters common to all particle 3D record comparison views (i.e. changing these settings for this view changes them for all particle views). Details regarding this dialog box are provided in section 9.2.5.1).
- *Draw point particles…* – a check menu item that determines whether the particles are displayed as points.
- *Draw true particles…* – a check menu item that determines whether the particles are displayed as three-dimensional spheres.

- *Show all particles* – when selected all particles, in both records, are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show record particles* – when selected all record particles with no corresponding particles in the comparison record are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show comparison particles* – when selected all comparison record particles with no corresponding particles in the record itself are displayed. This has the same effect as the equivalent button in the particle toolbar.
- *Show common particles* – when selected all particles common to both records are displayed. This has the same effect as the equivalent button in the particle toolbar.
- Show frame – when selected the frame drawn around the domain containing the particles is displayed.
- Show scales – when selected the scales on the axes are displayed. These scales are challenging to draw in three-dimensional space as they overlap one another, the axes themselves and the particles depending on the camera location and orientation. Hiding them may help to make the display easier to interpret.
- Show info – this item displays the temporal information that is located at the bottom of the view.
- Show regions – when selected the currently defined 3D regions are displayed.
- *Redraw during resize* – if set, the view redraws during a resize operation. If the drawing operation is extensive, for example many frames with large numbers of particles, this might cause a lack of responsiveness. If not set redrawing does not occur and the user needs to perform some manual operation after the resize in order to force a redraw.
- Change variable labels… – this item opens a dialog box in which the labels for the x, y, and z variables can be set. Selecting and right clicking the text in the text boxes enables the user to select Greek or Latin characters.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y, z and t variables can be set. Sensible minimum values are set for these.
- Reset min/max to default – this resets the x, y and z axis ranges to their defaults.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar. This can also be effected by double clicking the mouse while holding down the ALT key.

Animation Menu

- *Go to next time* – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.

- *Go to previous time* – steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Play* – starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Comparison Menu

- *Manage comparison…* – this option displays the comparison manager view - see section 9.2.5.2 for details.

Cameras menu

- *Manage cameras…* – opens the camera management view. See chapter 2 of the *User's Guide: Field Objects* manual.
- *Open current camera controller* – this opens the controller view of the currently selected camera so the user can make adjustments to the camera settings. These changes are often most easily achieved using keyboard commands (see chapter 2 of the *User's Guide: Field Objects* manual). The camera controller view is also opened by double clicking the mouse on the display area with SHIFT and ALT keys held down.

Lighting menu

- *Manage lighting…* – this option opens the manage lighting view. See chapter 3 of the *User's Guide: Field Objects* manual.

Region Menu

- *Manage regions…* – displays region 3D manager view (see chapter 48) and enables the user to create 3D regions in the view.
- *Load regions…*– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn3* extension.
- *Save regions as…*– saves all of the currently defined regions in a *.rgn3* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.

- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Delete particles in selected regions* – this will delete all particles that lie within the selected 3D regions for the frames currently displayed. The user will be asked to confirm the action.

Zoom Menu

- *Zoom* – causes the view to be zoomed using the currently displayed zoom rectangle.
- *Unzoom* – this undoes any zoom operation by redisplaying the entire video frame area.
- *Clear zoom rectangle*– this undoes any zoom operation by redisplaying the entire video frame area.

One popup menu is provided with this view. It contains items that replicate those in the main menu bar and is provided simply as an alternative means of accessing this functionality.

Popup Menu

- *Set layout…* – see View menu.
- *Go to next time* – see Animation menu.
- *Go to previous time* – see Animation menu.
- *Play* – see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show all particles* – see View menu.
- *Show record particles* – see View menu.
- *Show comparison particles* – see View menu.
- *Show common particles* –see View menu.
- Show frame – see View menu.
- Show scales – see View menu.
- Show info – see View menu.
- Show regions – see View menu.

9.2.5.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all comparison views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Number of x divisions* – this parameter specifies how many divisions appear along the x axis and hence how many scale values will be displayed.
- *Number of y divisions* – this parameter specifies how many divisions appear along the y axis and hence how many scale values will be displayed.
- *Number of z divisions* – this parameter specifies how many divisions appear along the z axis and hence how many scale values will be displayed.
- *Radius factor* – when the particles are drawn as spheres their radii are multiplied by this factor. This gives the user some control over how large the particles appear in the view.
- *Geometric scale factor* – this option allows the user to globally change the "size" of objects such as the axes thickness.
- *Font size* – sets the font size. The default is 14.
- *Draw axes at origin* – the default setting is to draw the axes to the left, bottom and back of the domain. By selecting this checkbox the axes are relocated so they pass through the origin.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *Particle colour* – the colour used for drawing all of the particles in the two records. It can be changed by clicking on the Select… button.
- *Record particle colour* – the used for drawing the original record only particles. It can be changed by clicking on the Select… button.
- *Comparison particle colour* – the colour used to draw the comparison record only particles . It can be changed by clicking on the Select… button.
- *Common particle colour* – the colour used to draw the common particles. It can be changed by clicking on the Select… button.
- *Background colour* – the background colour. It can be changed by clicking on the Select… button.
- *Use colour distance mapping* – when selected particles closer to the cameras are drawn brighter than those further away. This is only applicable when the particles are drawn as points.

9.2.5.2 Comparison management view

This view provides the user with a detailed breakdown of the particles in the record and the comparison record. The view includes 4 tabs – Statistics, Record, Comparison and Common.

Statistics

- *Particle record for comparison* – this drop down box lists all of the particle 3D records currently open in *Streams* that have the same number of frames. The user selects the record from this list that they wish to compare to the original record.
- *Tolerance* – this text box is used to enter the tolerance that determines whether two particles in the two records correspond to the same particle. Each particle in the original record is compared to all particles in the comparison record. The particle that lies closest to the original particles, and that is closer than the specified tolerance is designated as the same particle and added to the list of common particles. If no such particle exists then the particle is added to the list of record particles (that have no equivalent in the comparison record).
- *Number of record particles* – this is the number of particles (in all frames currently displayed) in the original record for which there are no corresponding particles in the comparison record.

- *Number of comparison particles* – this is the number of particles (in all frames currently displayed) in the comparison record for which there are no corresponding particles in the original record.
- *Number of common particles* – this is the number of particles (in all frames currently displayed) common to both records.
- *Compute statistics* – when this button is pressed the particle statistics are computed.

Record

- *Record particle list* – this list box displays all of the record particles (without corresponding particles in the comparison record) together with their location, size and colour.

Comparison

- *Comparison particle list* – this list box displays all of the comparison record particles (without corresponding particles in the original record) together with their location, size and colour.

Common

- *Common particle list* – this list box displays all of the common particles together with their location, size and colour.

Menus

View Menu

- Change decimal places… – this item opens a dialog box in which the number of decimal places used to display the data can be changed.

9.2.6 Diagnostic view

Description

The diagnostic view is intended to provide the user with a mechanism by which they can understand the impact of adopting different particle matching strategies on the matches in a particle record. Essentially the view computes comparison statistics between the matches for this record (termed the original record) and those for another record (termed the comparison record) that is assumed to contain exactly the same set of particles.

The view provides 3 tabs. These are described below.

Selection

This tab allows the user to select the comparison record with which the original record will be compared. In addition it allows the user to set certain analysis parameters.

- *Particle record for comparison* – this dropdown box lists all of the particle 3D records currently open in *Streams* that possess the same number of frames as this particle record. Each of these frames must possess exactly the same number of particles as this particle record's frames. The user selects the comparison record by choosing a record from this box. Typically the user will create a number of copies of the particle record of interest and analyse each of the records with a different set of PTV strategies. These records can be compared by ensuring that they are all open in *Streams* concurrently.
- *Compute diagnostic statistics* – the diagnostic statistics are computed once this button is pressed.

Matches

This tab provides the diagnostic statistics based on the matches in each frame. The data are presented in a table with the following columns.

- *Frame* – the number of the frame (starting at 0).
- *Particles* – the number of particles in the frame (must be the same for both records).
- *# of orig matches* – the number of the particles in the frame that are matched in the original record. This is also given as a percentage.
- *# of comp matches* – the number of particles in the frame that are matched in the comparison record. Also given as a percentage.
- *# of common matches* – the number of particles in the frame that are matched to the same particle in both of the particle records. Also given as a percentage.
- *# of conflict matches* – the number of particles in the frame that are matched to different particles in the two records. Also given as a percentage.
- *# of orig only matches* – the number of particles in the frame that are matched in the original record but not matched in the comparison record. Also given as a percentage.
- *# of comp only matches* – the number of particles in the frame that are matched in the comparison record but not matched in the original record. Also given as a percentage.

Velocities

This tab provides diagnostic statistics based on the particle velocities computed from the matches in each frame. In this view the symbol |du| refers to the magnitude of the difference in velocity between the predicted particle velocities based on the particle matches in each of the

records. If the two particle records have the same matches for a particle then this value will be 0. For records where particle displacements are the focus this velocity data can be converted to displacement data by multiplying by the time step between frames. The data are presented in a table with the following columns.

- *Frame* – the number of the frame (starting at 0).
- *Particles* – the number of particles in the frame (must be the same for both records).
- *Number of conflicts* – the number of the particles in the frame that are matched differently in the two records.
- Ave |du| all (unit) – |du| averaged over all particles in the frame. This is also given as a percentage of the particle velocity.
- Ave |du| conflict (unit) – |du| averaged over all particles that have conflicting matches in the frame. This is also given as a percentage of the particle velocity.
- Std |du| conflict (unit) – the standard deviation of |du| for all particles with conflicting matches.
- Max |du| conflict (unit) – the maximum value of |du| for the particles with conflicting matches.
- Min |du| conflict (unit) – the maximum value of |du| for the particles with conflicting matches.
- Num |du| conflict > 0 – the number of particles with conflicting matches for which the original particle record predicts a higher magnitude of the particle velocity.
- Num |du| conflict <= 0 – the number of particles with conflicting matches for which the original particle record predicts a lower or equal magnitude of the particle velocity.

Access

This view is accessed through the popup menu of a record. Particle records are displayed in the *Streams* object and particle record list views.

9.2.7 Particle list view

Description

The particle list view is designed to give the user access to particle information. The particle list view displays, in tabular form, details of the particles that are visible through a 2D rectangle drawn on screen. The rectangle is assumed to lie in the focal plane of the current camera. Details for each frame displayed in the record's particle view (up to a maximum of 10) are given on different tabs. The information displayed includes the particle state information if it is available. Right clicking

any particle in the list allows the user to display particle views (see chapter 7).

Access

This view is accessed through the popup menu for the 2D rectangle region in the particle view.

9.2.8 Match details view

Description

The match details view is designed to give the user access to internal details of any PTV 3D matching analyses that have been performed. The match details view displays, in tabular form, details of the particles that are seen through a 2D rectangular region drawn on the screen. Details for each frame displayed in the record's particle view (up to a maximum of 10) are given on different tabs. The information displayed includes the state information for each particle (number, location, radius and colour (RGB)), the particles to which each particle is matched in the previous and next frames, together with their state information, and the costs associated with these matches. These costs are calculated using a PTV pipeline defined for this particle record. If more than one pipeline is defined the user selects the desired one when the view is opened.

At the top of each tab a panel displays some basic statistics associated with the particles in the frame.

Access

This view is accessed through the popup menu for a 2D rectangle region in the particle view.

Menus

File Menu

- *Save to text file* – allows the user to save the tabulated data in this view to a text file (.txt or .csv).

9.3 Popup menu

The particle record provides the following options in its popup menu.

- *Save as…* – saves the particle record in a new file. All particle record files have the extension *.par3*. The new file name is displayed next to the particle record name in the *Streams* object list view.
- *Save* – saves the particle record to disk. If the particle record currently has no associated file then the user is prompted for a file name. All particle record files have the extension .par3.
- *Save Eulerian velocities to text file…* – saves the velocities of each particle in each frame to a comma (,) delimited text file (*.txt* or *.csv*). The user is at liberty to choose the frames that are to be included as well as whether the velocity can be calculated from a single match in the previous or next frame, or whether it requires matches in both frames. The data is output by frame.
- *Save Lagrangian velocities to text file…* – saves the velocities of each particle as it moves to a comma (,) delimited text file (*.txt* or *.csv*). Unlike the Eulerian data the Lagrangian data produces velocities along each particle path, instead of the velocities at each time step. By definition all velocities will be calculated from two matches, except for the first and last velocities in the path. These will have only a single match available. The data is output by particle path not by frame.
- *Save camera calibration data to text file…* – particle 3D records are created from particle 2D record bundles as described in the *System Theory and Design* manual. Perspective cameras used in bundles need to be calibrated using particle data for which the user has both the exact 3D particle locations as well as the 2D locations of the particles as seen by the camera. This option enables artificial calibration data to be created from a particle 3D record. A dialog box is opened that prompts the user to specify the frame number of the record to be used, the number of particles from the frame to output to file and the number of decimal places used to store particle information in the file. The user then also specifies a model camera (which may be created or loaded from file – see section 10.5). Once all information is selected the user specifies an output file and presses the OK button. The result is a text file used by perspective cameras for calibration – see chapter 10.
- *Create copy* – creates a copy of this particle record and adds it to the *Streams* system. The copied record includes all particle data but does not include the processes of the original particle record.
- *Change name…* – allows the user to change the name of the particle record. This name change will be reflected in the *Streams* object list view. This action can also be performed through the details view.
- *Edit description…* – allows the user to change the description of the particle record. This action can also be performed through the details view.
- *Delete frames…* – this option allows the user to delete frames from a particle record. The first and last frames to be deleted are selectable by the user. The deletion of frames will remove affected particle matches. The frames are renumbered after the deletion if appropriate. This action is drastic and the user is prompted for confirmation.
- *Edit frames…* – this option allows the user to thin the number of frames by retaining only 1 frame in every N where the user can specify the value of N. Most importantly if a particle in frame 1, that is to be retained, has a

particle path that extends to frame i+N then a match for this particle between frames I and i+N is created based on this path. Thus deleing frames in this way does not necessarily remove all matches. Clearly the larger the value of N the more likely that particle tracks will be lost. This action is drastic and the user is prompted for confirmation.

- *Compress domain…* – this option allows the user to compress the physical domain of the particle record. When a record is transformed extensions to its physical domain may be made. If the result of these transformations is such that there are regions in the particle record that contain no particles then the compress option will reduce the physical domain of the record to the smallest rectangle that contains all particles. The user is prompted as to whether they wish to re-zero the bottom left corner of the domain so that it corresponds to the origin.
- *Segment frames…* – this option allows the user to create a segmentation for the particle record. The user is prompted to specify the number of rows and columns in the segment array. Once the segmentation is created this menu option is ticked and selecting it again will remove the segmentation, thus freeing up the memory resources consumed by the segmentation.
- *Sort particles…* – this option allows the user to sort the particles within a frame. There are six submenus that specify how the particles are to be sorted. The particles can be order by increasing x, decreasing x, increasing y, decreasing y, increasing z or decreasing z. This only changes the particle numbering, no matches or other particle properties are changed.
- *Delete all matches…* – this option deletes all particle matches. The user is prompted for confirmation of the choice.
- *Delete all matched particles…* – this option allows the user to delete all particles within the record that are currently matched. The user is prompted for confirmation.
- *Delete all unmatched particles…* – this option allows the user to delete all particles within the record that are currently unmatched. The action user is prompted for confirmation.
- *Delete random particles…* – this option allows the user to delete a random selection of particles in the record. The user is prompted to specify the proportion of particles to delete (0-1). The process starts at frame 0 deleting the proportion of particles requested from that frame. If these particles are matched in subsequent frames the matched particles are deleted from those frames as well. Thus the number of new particles deleted in subsequent frames will be reduced based on the number of particles that have already been removed through matches to particles in previous frames. Note that as the particles selected to be deleted are chosen based on random numbers the exact number of particles removed is unlikely to match the requested proportion exactly. The user is prompted for confirmation of this action.
- *Create true matches* – (only available if the particle record has not been created from a theoretical model). This option allows the user to artificially create a set of true matches for particles in the particle record. The current matches are taken to be the true matches. The creation of

true matches will result in the particle record requiring more memory, both internal and on disk, for its storage.

- *Remove true matches* – (only available if the particle record has not been created from a theoretical model). This option allows the true matches created with the *Create true matches* option to be deleted.
- *Open details view* – opens the details view.
- *Open particle view* – opens the particle view. This is the default view.
- *Open process view* – opens the process view.
- *Open match summary view* – opens the match summary view.
- *Open diagnostic view* – opens the diagnostic view.
- *Open comparison view* – opens the comparison view.
- *Execute process pipeline* – executes the process pipeline if it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the record from *Streams*. All views are closed. If the record has unsaved data then the user is prompted to save this data before closure.

10 Particle 2D Record Bundles

10.1 Description

The particle 2D record bundles lie at the heart of *Streams'* three-dimensional PTV capability. In order to construct three-dimensional particle records current technology requires multiple two-dimensional imaging devices to be used to capture different perspectives of a three-dimensional scene. Combining the results from these multiple two-dimensional perspectives allows particle locations in 3D space to be inferred. The process by which these three-dimensional particles are constructed from bundles is described in chapters 28 and 29.

In this chapter we are concerned with the bundles themselves. A particle 2D record bundle contains a number of **perspectives**, where the number of perspectives matches the number of cameras being employed to observe the three-dimensional system of particles. Each perspective includes a particle 2D record and a **perspective camera**.

A particle 2D record bundle is not useful unless the perspective cameras are calibrated. The calibration of cameras capturing images for two-dimensional particle tracking analysis is simple. Generally it involves capturing the image of a ruler or some other artifact that enables a conversion from pixels to physical units to be made, or it involves the determination of suitable mapping functions that map pixel coordinates to spatial coordinates.

On the other hand, for perspective cameras capturing images for three-dimensional particle tracking the calibration is quite complex. To be able to interpret the two-dimensional images such a camera must be characterised in some way. As described in the *System Theory and Design* manual (see section 6.5) this characterization is implemented in *Streams* through two mapping functions that take three dimensional coordinates and map them onto and x and y coordinates in two-dimensional space.

To calibrate a perspective camera the user must have access to images of an array of particles whose three-dimensional coordinates are accurately known. Such an array is referred to as a **calibration target**. Once the corresponding two-dimensional coordinates of these target particles are known for a particular perspective camera the two mapping functions can be computed using a least squares fit. This process is executed

through the calibration view of the camera and it is robust and computationally inexpensive.

10.2 Views

The particle 2D record bundle provides two different views of the data it stores. These are:

- details view
- process view

10.2.1 Details View

Description

The details view provides access to the different perspectives that make up the bundle. Each perspective includes a particle 2D record and a perspective camera.

Note that all particle records in the bundle must have the same number of frames and the same time step.

The user is at liberty to add and remove perspectives from the bundle and change other bundle details. The user must press the Save button for these changes to take effect. If the changes are to be discarded press the Cancel button.

The view has three tabs, Details, Description, and Perspectives. These are described below.

Details

- *Bundle name* – the name associated with the bundle.
- *Number of frames* – this is the number of frames each particle record possesses. It cannot be editted by the user. If the bundle includes no perspectives it is set to -1.
- *Temporal unit* – this temporal unit for the bundle. In theory all of the particle records should share this unit but this is not enforced.
- *Spatial unit* – the user defined spatial unit. In theory all of the particle records should share this unit but this is not enforced.
- *Time step (unit)* – the time step each particle record possesses. It cannot be editted by the user. If the bundle includes no perspectives it is set to -1.

Description

- *Description* – the user can provide a description of the bundle in the text box provided.

Perspectives

- *Perspective list* – this list box at the top of the tab lists all of the currently defined perspectives in the bundle. Right clicking on one of these will display its popup menu and double clocking will open the details view of the perspective.
- *Load…* – loads a perspective from file and adds it to the list. The loaded perspective must have the same number of frames and the same time step as the perspectives that already exist. A perspective file has a *.par2per* extension.
- *Add…* – creates a perspective and adds it to the list. A dialog box is displayed that enables the user to load a particle 2D record from file and either load a perspective camera from file or create a new one. The new perspective must have the same number of frames and the same time step as the perspectives that already exist.
- *Delete* – deletes the currently selected perspective. If no perspectives exist after this action the number of frames and time step for the bundle are set to -1.

Access

This view is accessed through the popup menu of a particle 2D record bundle or by double clicking the bundle. Particle 2D records bundles are displayed in the *Streams* object and particle record bundle list views.

10.2.2 Process view

Description

The process view for a particle record is exactly the same as that for an image sequence. See section 3.2.3.

10.3 Popup menu

The particle 2D record bundle provides the following options in its popup menu.

- *Save as…* – saves the particle 2D record bundle in a new file. All bundles have the extension *.par2bun*. The new file name is displayed next to the bundle name in the *Streams* object list view.

- *Save* – saves the particle 2D record bundle to disk. If the bundle currently has no associated file then the user is prompted for a file name. All bundles have the extension .par2bun.
- *Change name*… – allows the user to change the name of the particle 2D record bundle. This name change will be reflected in the *Streams* object list view. This action can also be performed through the details view.
- *Edit description*… – allows the user to change the description of the bundle. This action can also be performed through the details view.
- *Open details view* – opens the details view.
- *Open process view* – opens the process view.
- *Execute process pipeline* – executes the process pipeline if it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the bundle from *Streams*. All views are closed. If the bundle has unsaved data then the user is prompted to save this data before closure.

10.4 Perspectives

A perspective encapsulates the information that uniquely defines one view of a three-dimensional system of particles. This includes the camera that defines the perspective together with the particle 2D record that is obtained when the camera records the three-dimensional particles.

A perspective provides one view of its data. It is:

- details view

10.4.1 Details View

Description

The details view lists the particle 2D record and the perspective camera for this perspective and enables the user to access them through their popup menus. The view also allows the user to change the particle record or camera by loading a new one, or, in the latter case, creating a new one.

The user must press the Save button to save changes made to the perspective. If the changes are to be discarded then the user should press the Cancel button.

The view has two tabs, Name, and Camera and Record. These are described below.

Name

- *Name* – the name of the perspective that will appear in the perspective list of the bundle.

Camera and Record

- *Particle 2D record* – this list box lists the particle 2D record that belongs to this perspective. The particle 2D record can be manipulated through its popup menu or by double clicking it.
- *Load record* – this button allows a new record to be loaded from file to replace the one listed. The user is asked to confirm that they wish to replace the current record.
- *Camera* – this list box lists the perspective camera that belongs to this perspective. The camera can be manipulated through its popup menu or by double clicking it.
- *Load camera* – this button allows a new perspective camera to be loaded from file to replace the one listed. The user is asked to confirm that they wish to replace the current camera.
- *Create camera* – this button allows the user to create a new perspective camera to replace the one currently listed. The user is asked to confirm the choice.

Access

This view is accessed through the popup menu of a perspective or by double clicking it.

10.4.2 Popup menu

The perspective provides the following options in its popup menu.

- *Save as…* – saves the perspective in a new file. All perspectives have the extension *.par2per*.
- *Open details view* – opens the details view.
- *Close all views* – closes all open views of the perspective.

10.5 Perspective cameras

A perspective camera is the means by which *Streams* can map particles in three-dimensional space onto the two-dimensional image recorded by the camera and stored in a particle 2D record. One particle 2D record is not enough to produce three-dimensional particle fields, but with a number of such records, and their associated cameras, three-dimensional particle fields can be constructed.

A perspective camera is characterized by two mapping functions that map coordinates in three-dimensional space to the x and y coordinates of the camera's two-dimensional projection. Thus, given any cluster of particles in three-dimensional space the mapping functions can reconstruct the two-dimensional scene recorded by the camera. These mapping functions do not have uniquely defined inverses and it is for this reason that three-dimensional scenes can only be reconstructed through the combined imaging of multiple cameras.

A brief overview of the calibration of perspective cameras is provided at the beginning of this chapter.

A perspective camera provides two different views of the data it stores. These are:

- details view
- calibration view

10.5.1 Details View

Description

The details view provides access to the all of the camera information including the mapping functions that define it. In theory, the mapping functions could be specified manually. But, in practice, these functions are determined by a calibration process whereby the image of a known calibration target is used to compute the mapping function parameters (the polynomial coefficients). This process is executed using the calibration view.

The mapping functions are *n*th order polynomials in three dimensions and as such the number of coefficients, *N*, in the polynomials is given by

$$N = \frac{n\left(n^2 + 6n + 11\right)}{6} + 1 \qquad (10.1)$$

The user must press the Save button to save changes made to the parameters. If the changes are to be discarded then the user should press the Cancel button.

The view has three tabs, Camera, Domain, and Coefficients. These are described below.

Camera

- *Camera name* – the name that will appear in the camera list for the perspective.
- *Aspect ratio* – the aspect ratio of the camera – height/width – can be computed from the pixel resolution of the camera. This parameter is entered by the user. It is not computed through the calibration process.
- *Polynomial order* – the order of the polynomials used to characterize the camera. This cannot be changed in this view. It is specified in the calibration view.

Domain

- *x min* – the minimum value of x found in the calibration target data. This cannot be editted by the user.
- *x max* – the maximum value of x found in the calibration target data. This cannot be editted by the user.
- *y min* – the minimum value of y found in the calibration target data. This cannot be editted by the user.
- *y max* – the maximum value of y found in the calibration target data. This cannot be editted by the user.
- *z min* – the minimum value of z found in the calibration target data. This cannot be editted by the user.
- *z max* – the maximum value of z found in the calibration target data. This cannot be editted by the user.

Coefficients

- *Table of coefficients* – the table of coefficients summarises the polynomial mapping functions computed through the calibration process. The table has 6 columns. These are: i – this is the coefficient number; x mapping – the polynomial coefficient for the x mapping; y mapping – the polynomial coefficient for the y mapping; x^ - the power of x in this term in the mapping function; y^ - the power of y in this term in the mapping function; z^ - the power of z in this term in the mapping function.

Access

This view is accessed through the popup menu of a perspective camera.

Menus

View Menu

- *Change decimal places…* – allows the user to specify the number of decimal places used to display the polynomial coefficients in this view. This does not affect the underlying accuracy of the coefficients.

10.5.2 Calibration view

Description

The calibration view enables the user to perform a calibration of a perspective camera. This calibration uses the calibration target particle information to deduce the coefficients in the two mapping functions. The crucial information that must be provided for the calibration process is a set of particle data. These data include both the known three-dimensional locations of each particle and the two-dimensional coordinates of the particles obtained from an analysis of an image of the particles captured by the camera. Note that these calibration particle data may bear no relationship to the particle data in the particle 2D record associated with the camera. That particle record is likely to contain images of the flow of interest, not the calibration particles.

The calibration process attempts to minimize the error between the measured 2D particle locations and those computed from the 3D coordinates using the mapping functions.

A sensible approach to test the robustness of the calibration is to use a subset of the available calibration particles to deduce the mapping functions and then to use the remaining calibration particles as a check.

The view has three tabs, Parameters, Calibration points, and Error fields. These are described below.

Parameters

- *First calibration point* – the number of the point (particle) in the list of calibration points that is the first to be used for calibration purposes.
- *Number of calibration points* – the number of points (particles), starting from the first, to be used in the calibration. The calibration points must be sequential in the list.
- *First check point* – the number of the first point to be used for checking the calibration.
- *Number of check points* – the number of points (particles), starting from the first, to be used in the check. The check points must be sequential in the list.
- *Polynomial order* – the order of the polynomial to be used in the mapping functions.

Calibration points

- *Calibration points – this table lists all of the calibration points available to the calibration and checking algorithms. The table includes 8 columns. These are: Number – the number of the point in the list; Mapped x – the x coordinate of the particle in two dimensional space; Mapped y – the y coordinate of the particle in two dimensional space; Original x – the x coordinate of the particle in three dimensional space; Original y – the y coordinate of the particle in three dimensional space; Original z – the z coordinate of the particle in three dimensional space; Calibration error – the error computed as the distance, in 2D space, between the mapped point and the original point mapped to 2D space using the camera mapping functions, if this point is used for the calibration; Check error – the error computed as the distance, in 2D space, between the mapped point and the original point mapped to 2D space using the camera mapping functions, if this point is used as a check.*
- *Calibrate* – this button initiates the calibration process. The calibration points specified in the Parameters tab are used to compute the polynomial coefficients of the two mapping functions. An error field will be produced and added to the list in the Error fields tab.
- *Check* – this button initiates the check routine. The check points specified in the Parameters tab are used in the check calculations. An error field will be produced and added to the list in the Error fields tab.
- *Calibration error* – the rms error for all points used in the calibration.
- *Check error* – the rms error for all points used in the check.
- *Load points* – this button loads calibration points from a file and adds them to the table of calibration points. The data should be in a space delimited .txt or .prn file. The file should have the following format. The first line should have a single integer value corresponding to the number of calibration points in the file. Each succeeding line should have 8 space delimited numbers. The first is the particle number (starting at 0), the next two are the mapped x and y coordinates followed by the particle radius. The last four are the original x, y, z coordinates and the radius of the particle in 3D space. Currently the two radii are not used in the calibration process and can take any value. The user will need to give some thought to the number of decimal places used for the two coordinate sets as these may limit the accuracy of the calibration routine. It is expected that the mapped x and y coordinates are normalized so that the maximum x value is 1 and the maximum y value is the aspect ratio of the image (number of vertical pixels/number of horizontal pixels)
- *Edit* – allows the user to edit the selected calibration point in the table. A dialog box is displayed to enable this.
- *Add* – allows the user to add a calibration point. A dialog box is displayed in which the user can enter the mapped and original coordinates. The new point is added to the end of the list.
- *Insert* – allows the user to insert a calibration point before the currently selected point in the table. A dialog box is displayed in which the user can enter the mapped and original coordinates.
- *Delete* – allows the user to delete a calibration point. The user is asked to confirm the action.

- *Clear* – allows the user to clear all calibration points. The user is asked to confirm the action.
- *Randomise* – allows the user to randomly reorder the calibration points in the table. As the calibration and check points must be sequential this can be a useful way of ensuring a random selection of points is used for the calibration and similarly for the check.
- *Add 2D noise* – allows the user to add noise to the two dimensional particle coordinates. This may be useful in analyzing the robustness of the calibration result. The user is prompted for a noise level (this is a fractional noise level for each coordinate – i.e. 0.01 would be a 1% error based on the maximum x or y coordinate) and then random noise with a maximum error corresponding to the level set by the user is applied to the x and y coordinates of the mapped particles.
- *Add 3D noise* – allows the user to add noise to the three dimensional particle coordinates. This may be useful in analyzing the robustness of the calibration result. The user is prompted for a noise level (this is a fractional noise level – i.e. 0.01 would be a 1% error based on the maximum x , y or z coordinate) and then random noise with a maximum error corresponding to the level set by the user is applied to the x, y and z coordinates of the calibration particles.
- *Shift 3D points* – allows the user to shift the x and y coordinates of the three dimensional particles a specified amount. The shift in the two directions is specified by the user in a dialog box and is an absolute value. The user is also able to specify a range of z coordinates that limits the particles to which this shift is applied. All particles with a z coordinate within the specified range are shifted. This option is again designed to enable the exploration of the impact of particle coordinate errors on the camera calibration. The calibration particles typically used in our laboratory are planes of 2D particles with accurately known 2D coordinates that are moved in the z direction to provide a full 3D calibration target. Errors in the accuracy of movement in the z direction may result in the particle x and y coordinates being slightly shifted as they are moved.
- *Restore camera…* – this button restores the mapping functions that existed before the last calibration process was initiated. The user is prompted to confirm this action.
- *Save as Perspective…* – pressing this button saves the current camera settings and the mapped 2D points as a perspective. The user is prompted for the file used to store the perspective. This can be a useful tool in reconstructing the three dimensional target using images from a number of calibrated cameras.

Error fields

- *Error fields (deleted on view closure)* – to provide the user with a picture of the error distribution in the mapped two-dimensional domain a two-dimensional vector field is produced for each calibration or check operation. The vector field indicates the displacement between each mapped point and its computed equivalent. These fields are deleted once the view closes.

- *Delete field* – the user can manually delete error fields from the list by pressing this button. The fields selected in the list are deleted. The user is asked to confirm the action.

Access

This view is accessed through the popup menu of a perspective camera or by double clicking the camera.

Menus

View Menu

- *Change decimal places…* – allows the user to specify the number of decimal places used to display data in this view.

10.5.3 Popup menu

The perspective camera provides the following options in its popup menu.

- *Save as…* – saves the camera in a new file. All perspective cameras have the extension *.percam*.
- *Open details view* – opens the details view.
- *Open calibration view* – opens the calibration view.

10.6 Model cameras

Prior to *Streams 3.02* perspective cameras and their operating environments were characterized in the traditional way through a number of geometric and optical parameters. A camera was described by its location (in 3D space), its orientation, its zoom angle and its aspect ratio. In addition, any lens distortion was catered for through a general lens distortion model (described below). The camera's environment enabled a simple experimental setup to be modelled. This setup included the presence of three optical media (that in which the camera was located, the flow under consideration and a straight, parallel-sided partition between them) all with different refractive indices.

While this model of a perspective camera has been replaced by the mapping functions, such a physically characterized camera is still useful in creating artificial data sets with which to explore *Streams'* three-dimensional capabilities. Such a camera is designated as a **model camera**.

Before providing a description of how model cameras can be used the lens distortion model is briefly explained. The lens distortion is computed in the following way. Consider a coordinate system with an origin at the centre of the two-dimensional view of the camera and normalized by the view width. In such a coordinate system x will vary from -0.5 to 0.5 and y will vary from -0.5AR to 0.5AR where AR is the aspect ratio. If (x,y) are the undistorted coordinates of a point relative to that coordinate system then the lens distortion will cause these coordinates to be mapped to the distorted coordinates (x',y') given by

$$x' = x\left(1+K_1 r+K_2 r^2+K_3 r^3\right) + P_1\left(r^2+2x^2\right) + 2P_2 xy \qquad (10.2)$$

$$y' = y\left(1+K_1 r+K_2 r^2+K_3 r^3\right) + 2P_1 xy + P_2\left(r^2+2y^2\right) \qquad (10.3)$$

Model cameras play a useful role in the creation of artificial datasets that may be used to investigate the various aspects of *Streams*' three-dimensional capabilities. Section 10.8 describes a process whereby a particle 2D record bundle can be created from a particle 3D record using a suite of model cameras.

10.6.1 Details View

Description

The details view provides access to the all of the model camera and environmental parameters described above.

The user must press the Save button to save changes made to the parameters. If the changes are to be discarded then the user should press the Cancel button.

The view has four tabs, Camera, Geometry, Lens distortion, and Media. These are described below.

Camera

- *Camera name* – the name that will appear in any camera list.

Geometry

- *Location (x,y,z)* – the coordinates in 3D space of the camera, taken as being located at a single point.

- *Projection direction (x,y,z)* – a unit vector pointing in the direction from the camera to its focal point.
- *Up direction (x,y,z)* – a unit vector that indicates which direction the camera views as up. It is normal to the projection direction.
- *Horizontal zoom angle* – the horizontal zoom angle of the camera. This determines the extent of the camera's view. It is given in degrees.
- *Aspect ratio* – the aspect ratio of the camera – height/width.

Lens distortion

- *K1* – the parameter appearing in equations 10.2 and 10.3.
- *K2* – the parameter appearing in equations 10.2 and 10.3.
- *K3* – the parameter appearing in equations 10.2 and 10.3.
- *P1* – the parameter appearing in equations 10.2 and 10.3.
- *P2* – the parameter appearing in equations 10.2 and 10.3.

Media

- *Use multimedia* – this box is ticked when the camera is viewing the particles through three different media as described above. If the camera is viewing through one medium alone then this box should be left unchecked.
- *Refractive index 1 (camera)* – the refractive index of the medium in which the camera resides – normally air.
- *Refractive index 2 (partition)* – the refractive index of the medium which separates the camera and the flow – normally the flow tank material's refractive index.
- *Refractive index 3 (flow)* – the refractive index of the fluid under observation.
- *Partition thickness* – the thickness of the partition in appropriate physical units.
- *Partition equation (a,b,c,d)* – the orientation of the partition in space is important. The partition is assumed to be a plane and the orientation of the face closest to the camera is expressed by the equation ac + by + cz + d = 0. It is assumed that the expression on the left hand side of the equation decreases as a point moves from the face towards the camera. So for example if the partition is assumed to be in the z = 0 plane with the z axis pointing towards the camera, this equation would be –z = 0.

Access

This view is accessed through the popup menu of a model camera or by double clicking the camera.

10.6.2 Popup menu

The model camera provides the following options in its popup menu.

- *Save as…* – saves the model camera in a new file. All model cameras have the extension *.modcam*.
- *Open details view* – opens the details view.

10.7 Creating particle 2D record bundles

Particle 2D record bundles are created through the Create particle 2D record bundle… menu option in the Particle menu of the *Streams* application view. This command opens a dialog box in which the user can construct the bundle. The dialog box closely matches the details view of the bundle.

The dialog box has three tabs, Details, Description, and Perspectives. These are described below.

Details

- *Bundle name* – the name associated with the bundle.
- *Number of frames* – this is the number of frames each particle record possesses. It cannot be edited by the user. If the bundle includes no perspectives it is set to -1.
- *Temporal unit* – this temporal unit for the bundle. In theory all of the particle records should share this unit but this is not enforced.
- *Spatial unit* – the user defined spatial unit. In theory all of the particle records should share this unit but this is not enforced.
- *Time step (unit)* – the time step each particle record possesses. It cannot be edited by the user. If the bundle includes no perspectives it is set to -1.

Description

- *Description* – the user can provide a description of the bundle in the text box provided.

Perspectives

- *Perspective list* – this list box at the top of the tab lists all of the currently defined perspectives in the bundle. Right clicking on one of these will display its popup menu and double clocking will open the details view of the perspective.

- *Load…* – loads a perspective from file and adds it to the list. The loaded perspective must have the same number of frames and the same time step as the perspectives that already exist. A perspective file has a *.par2per* extension.
- *Add…* – creates a perspective and adds it to the list. A dialog box is displayed that enables the user to load a particle 2D record from file and either load a perspective camera from file or create a new one. The new perspective must have the same number of frames and the same time step as the perspectives that already exist.
- *Delete* – deletes the currently selected perspective. If no perspectives exist after this action the number of frames and time step for the bundle are set to -1.

10.8 Generate particle 2D record bundle process

Three-dimensional particle tracking is a considerably more complex process than its two-dimensional counterpart. In order to help the user come to grips with the various aspects of a particle 2D record bundle and its behavior, without having to undertake complex experimental measurements, a particle 3D record can create an artificial bundle through a generate particle 2D record bundle process. This process is created in the particle record's process view.

The user creates or loads a number of model cameras that map the three-dimensional particles in the particle record onto two-dimensional space. From these a particle 2D record bundle is created containing one perspective for each model camera. Each perspective contains the particle 2D record generated by its associated model camera together with a perspective camera ready to be calibrated. The user can then explore the calibration process for the perspective cameras as well as the reconstruction of the particle 3D record from the bundle using the processes described in chapters 28 and 29.

Synthetic calibration data is readily created through the Save camera calibration data to text file… menu option in the particle 3D record's popup menu. This command generates the calibration text file required by the calibration view of a perspective camera. The user selects a number of parameters including a model camera that will generate the two-dimensional projections of the particle data in the particle 3D record. The record used can be the one from which the bundle is generated but it need not be. Instead it could be an artificially created calibration target produced using the free process described in section 22.4.

In order to simulate more complex experimental configurations the user can specify a number of three-dimensional regions in the generation process. These regions are treated as "obstacles" in the flow and, depending on their characteristics, may prevent some cameras from observing some of the particles in the particle 3D record. The reconstruction of the three-dimensional particle field then becomes more challenging due to these obscured views.

The generate particle 2D record bundle process has a single view.

10.8.1 Details view

Description

The details view enables the user to enter all of the information needed to produce a particle 2D record bundle.

The Save and Cancel buttons at the bottom of the view allow the user to save changes or abandon them.

The view has four tabs, Process, Details, Cameras and Regions. These are described below.

Process

- *Process name* – the name of the process.

Details

- *Bundle name – the name to be given to the bundle produced.*
- *Temporal unit* – the temporal unit for the bundle.
- *Spatial unit* – the spatial unit for the bundle.
- *All frames* – if checked all frames in the particle 3D record will be mapped into the resultant particle 2D records. If not checked the user can specify which frames to use in the next two parameters.
- *First frame* – if the All frames box is not checked this is the first frame of the particle 3D record that will be mapped into the particle 2D records.
- *Last frame* – if the All frames box is not checked this is the last frame of the particle 3D record that will be mapped into the particle 2D records.

Cameras

- *Model cameras* – to produce each particle 2D record in the resulting bundle a model camera must be specified. This box lists all of the model cameras that will be used when the process is executed. The resulting

number of perspectives in the bundle (i.e. particle 2D record and perspective camera) will match the number of model cameras in this list.

- *Load* – loads a model camera from file and adds it to the list. The file will have a *.modcam* extension.
- *Add* – adds a model camera to the list. A dialog box is displayed to allow the user to enter the camera parameters.
- *Delete* – deletes the selected cameras in the list. The user is asked to confirm their action.

Regions

- *Regions present* – a list of 3D regions (see chapter 48) that are treated as obstacles in the flow. Cameras cannot see through these obstacles.
- *Load* – loads a 3D region from a file and adds it to the list. The file will have a *.rgn3* extension.
- *Delete* – deletes the regions in the list. The user is asked to confirm their action.

Access

This view is accessed through the popup menu of a generate particle 2D record bundle process.

10.8.2 Popup menu

The generate particle 2D record bundle provides the following options in its popup menu.

- *Save as…* – saves process to file.
- *Lock* – locks the process so changes cannot be made inadvertently.
- *Change name…* – allows the user to change the process name.
- *Open details view* – opens the details view.
- *Execute* – executes the process.
- *Close all views* – closes all open views.
- *Close* – closes the process.

11 Intensity Field Time Series

11.1 Description

An intensity field time series is an Eulerian field that represents a time series of two-dimensional intensity fields calculated from the images in an image sequence. These intensities can be calculated directly from the original images, or from the output of some process applied to the images. Depending on the type of data that is available the user can select the intensity to be used. For example for an RGB image the intensity can be any of the three colour intensities, or an average of all three.

An intensity field is created by the create intensity field process operating on an image sequence (see chapter 30). Intensity field transform pipelines can create new intensity fields from already existing fields (see chapter 43).

An intensity field includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described in section 11.4.

Once created, an intensity field object is displayed in the *Streams* object and Eulerian field list views.

11.2 Views

The intensity field provides three views. These are

- details view
- calculator view
- process view

11.2.1 Details view

Description

The details view gives information about the intensity field. This information includes its name and description, details about the spatial grid on which the intensity field is defined, the time step between fields

and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the intensity field object.

11.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the intensity field. Currently seven calculators are available and these are discussed in detail in section 11.4.

Access

Access to the calculator view is through the popup menu of the intensity field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

11.2.2.1 Custom calculator dialog

When the user chooses to create a custom calculator a dialog box displayed. It contains the following tabs:

Name

On this tab the user can specify the name of the calculator. If the user leaves this field blank the formula text is substituted for the name. This name is displayed in the calculator list in the calculator view.

Formula

A text area is provided for the user to enter the formula for the calculator. This formula is defined by the rules described in chapter 30.

11.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the intensity field object.

11.3 Popup menu

The intensity field provides the following options in its popup menu.

- *Save as…* – saves the intensity field in a new file. All intensity field files have the extension *.int*. The new file name is displayed next to the intensity field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the intensity field to disk. If the intensity field currently has no associated file then the user is prompted for a file name. All intensity field files have the extension *.int*.
- *Change name…* – allows the user to change the name of the intensity field. This name change will be reflected in the *Streams* object and Eulerian field list views.
- *Edit description…* – allows the user to change the description of the intensity field.
- *Delete fields…* – this option allows the user to delete fields within the intensity field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the intensity field on disk. To save space it may be convenient to delete all of the calculated fields before saving the intensity field to disk.

- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the intensity field from *Streams*. All views are closed. If the intensity field has unsaved data then the user is prompted to save this data before closure.

11.4 Calculators

Each intensity field contains a number of predefined calculators that allow the user to extract information of interest from the intensity field. Each calculator produces a three dimensional field derived from the original intensity field. The available calculators are shown in table 11.1.

Table 11.1 The list of available density field calculators.

Symbol	Description
I	the instantaneous intensity field
I'	the fluctuating intensity field. The time-averaged intensity, calculated from the entire time series, is subtracted from I to generate this field.
I'I'	the intensity of the fluctuating intensity.
grad(I)	the gradient of the intensity field
|grad(I)|	the magnitude of the gradient of the intensity field
$\partial I/\partial x$	partial derivative of intensity with respect to x
$\partial I/\partial y$	partial derivative of intensity with respect to y

11.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. The field calculator manager view allows the user to calculate any one of the 30 different projections that the field calculator provides. All field calculators have the same manager view, and therefore this view will be described in a generic way.

Each field calculator generates a derived three-dimensional field from the intensity field object for which it is defined. The user is at liberty to extract a range of different projections of that three-dimensional field by

specifying constraints on the three independent variables x, y and t. For each variable the user can select

- to leave that variable as a variable in the calculated field, specifying the range for that variable, or
- to average over that variable, specifying the range of that variable over which the average is to be calculated, or
- to choose a value for that variable, at which the field will be calculated.

In addition, the user can choose to view the field along a line in the x-y plane. This line is known as a **cut**. The user specifies the end points of the cut and number of points that lie along the cut. In general the cut points will not coincide with grid points and the field is interpolated onto these points. The time variable may still be specified as variable, averaged or constant for a cut field. Thus the user has a total of 30 different options.

A set of radio buttons, text boxes and drop down boxes allow the user to select the projection of interest. Once these controls have been set the field is calculated when the Calculate field button is pressed. As the calculation of some fields can take some seconds the computation is run in a separate thread so that the user can perform other tasks if required. Progress of the calculation is provided in a monitor dialog. Once calculated the field is displayed in the list box labelled Calculated fields. Each field is given a unique name and an icon that indicates the type of field it represents (scalar, or vector, 0, 1, 2 or 3 dimensional).

The fields are given two names that indicate their genesis. The first, short name, matches the name of the calculator itself. The name of the second, longer name, begins with the field calculator name but this is followed by parentheses containing three groups of symbols, separated by commas, referring to the variables, x, y and t respectively (except in the case of a cut). These symbols are

- the variable name, if that variable was chosen to be variable, followed by its range, or
- the variable name in angle brackets (<>) if that variable was averaged, followed by the limits for the averaging process, or
- the variable name, followed by its value, if that variable was fixed, or
- for a cut the symbol "xy" is used followed by two ordered pairs specifying the start and end points of the cut.

Finally the name is completed with the name of the intensity field time series preceded by the word "from".

One or more fields can be selected and deleted by clicking the Delete field button, or they can be closed using the field object's popup menu.

The Calculated fields list supports the drag action so that fields in this list can be dragged (by clicking and holding the left mouse button and moving the mouse) to a list box that supports the drop action for fields. An example is the list of fields in the analysis view of a tablet (see User's Guide: Field Objects chapter 12). Some field views support the display of overlay fields and these views have the drop capability.

When an intensity field object is saved to disk all of its calculators and their calculated fields are saved with it. As these fields can be easily recalculated, if storage space is at a premium, the user may wish to delete all fields before saving the intensity field object.

Access

The manager view is opened by double clicking on the field calculator in the intensity field calculator view, or through the popup menu of the field calculator object.

Popup Menu

When a field object in the Calculated field list is right clicked its popup menu is displayed, and when it is double clicked its default view is opened. If multiple field objects are selected a right click of the mouse will display a separate popup menu with the following options:

- *Open default views* – this option will open the default view for each of the selected field objects.
- *Close all views* – this option will close all of the views of the selected field objects.

11.4.2 Custom calculators

Intensity fields also provide the user with the option to create their own custom calculator based on a mathematical formula (see chapter 49). A custom calculator, that can compute scalar fields, is created through the calculator menu of the calculator view.

11.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

12 Velocity 2D Field Time Series

12.1 Description

A velocity 2D field time series is an Eulerian field that represents a time series of two-dimensional velocity fields that are calculated from the frames in a particle 2D record. It can be created in a number of ways, using either a create velocity field process (see chapter 31), one of the free processes that combine Eulerian fields (see chapter 22) or a velocity 2D field transform pipeline (see chapter 43).

A velocity 2D field time series includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described in detail later in this chapter.

Once created, a velocity 2D field time series object is displayed in the *Streams* object and Eulerian field list views.

12.2 Views

The velocity 2D field time series provides three views. These are

- details view
- calculator view
- process view

12.2.1 Details view

Description

The details view gives information about the velocity field. This information includes its name and description, details about the spatial grid on which the velocity field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the velocity field object.

12.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the velocity field. Currently 30 calculators are available and these are discussed in detail in section 12.4.

Access

Access to the calculator view is through the popup menu of the velocity field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

12.2.2.1 Custom calculator dialog

When the user chooses to create a custom calculator a dialog box displayed. It contains the following tabs:

Name

On this tab the user can specify the name of the calculator. If the user leaves this field blank the formula text is substituted for the name. This name is displayed in the calculator list in the calculator view.

Formula

For a scalar field calculator a single text area is provided for the user to enter the formula for the calculator. This formula is defined by the rules

described in chapter 46. For a vector field calculator two text areas are provided for entering the formulae for the x and y components of the field.

12.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the velocity field object.

12.3 Popup menu

The velocity field provides the following options in its popup menu.

- *Save as*… – saves the velocity field in a new file. All velocity field files have the extension *.vel2d*. The new file name is displayed next to the velocity field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the velocity field to disk. If the velocity field currently has no associated file then the user is prompted for a file name. All velocity field files have the extension .vel2d.
- *Save* fields to text file …– saves the velocity field to disk as a text file. The user is presented with a dialog box in which they can choose the file, the number of decimal places for the field values, and whether derivative fields should be stored as well. The resulting text file has the following comma delimited format. In the first line is the size of the field, the number of time steps, and the number of grid points in the x and y directions. Then for each time step data is presented in the following way. The time is output on the first line. Then for each field that is output (this could be just u and v or include the 4 derivative fields, in order $\partial u/\partial x$, $\partial u/\partial y$, $\partial v/\partial x$, $\partial v/\partial y$) a table is written to the file. The first row is the x coordinates of each grid point. In the following rows the first entry is the y coordinate followed by the field value at the corresponding grid point. The string NAN is output if the field is undefined.
- *Change name*… – allows the user to change the name of the velocity field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description*… – allows the user to change the description of the velocity field.

- *Delete fields…* – this option allows the user to delete fields within the velocity field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the velocity field on disk. To save space it may be convenient to delete all of the calculated fields before saving the velocity field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the velocity field from *Streams*. All views are closed. If the velocity field has unsaved data then the user is prompted to save this data before closure.

12.4 Calculators

Each velocity field contains a number of predefined calculators that allow the user to extract information of interest from the velocity field. Each calculator produces a three dimensional field derived from the original velocity field.

The available calculators are shown in table 12.1.

Each of these calculators has a single view, the manager view. This view provides the user with control over which projection of the calculated field they wish to compute.

The manager view is common to all Eulerian 2D field calculators and is discussed in detail in section 11.4.1.

Table 12.1 The list of available velocity field calculators.

Symbol	Description
(u,v)	the instantaneous velocity vector
(u',v')	the fluctuating velocity vector. The time averaged velocity vector, calculated from the entire time series is subtracted from (u,v) to generate this vector.
|uv|	the magnitude of the instantaneous velocity vector
u	the x component of the instantaneous velocity vector
v	the y component of the instantaneous velocity vector
(∂u/∂t, ∂v/∂t)	the local acceleration vector
coverage	the coverage is defined to be 1 at a grid point at which the velocity is defined and 0 if it is not defined.
KE	the 2D kinetic energy based on the instantaneous velocity field.
ω	the vorticity field
div(u,v)	the 2D divergence of the instantaneous velocity field
strain rate	the strain rate of the instantaneous velocity field
TKE	the 2D turbulent kinetic energy. The time average of this field gives the 2D TKE per unit mass.
-u'v'	the negative correlation between the fluctuating velocity components. The time average of this field gives the turbulent shear stress per unit mass
u'u'	the intensity of the fluctuating x velocity component.
v'v'	the intensity of the fluctuating y velocity component.
u'u'u'	triple velocity fluctuation correlation.
u'u'v'	triple velocity fluctuation correlation.
u'v'v'	triple velocity fluctuation correlation.
v'v'v'	triple velocity fluctuation correlation.
u'	the x component of the fluctuating velocity field
v'	the y component of the fluctuating velocity field
∂u/∂x	partial derivative of u with respect to x.
∂u/∂y	partial derivative of u with respect to y.
∂v/∂x	partial derivative of v with respect to x.
∂v/∂y	partial derivative of v with respect to y.

uT	the barotropic velocity. This is the depth (vertically) averaged horizontal velocity component. It is uniform in the vertical.
uC	the baroclinic velocity. The difference between the instantaneous horizontal velocity and the barotropic velocity.
(uT,0)	the barotropic velocity in vector form with a zero vertical velocity.
(uC,0)	the baroclinic velocity in vector form with a zero vertical velocity.
(uC,uT)	a vector whose two components are the baroclinic and barotropic velocity components.

12.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

12.4.2 Custom calculators

Velocity fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49). See section 11.2.2 for more details.

12.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

13 Velocity 3D Field Time Series

13.1 Description

A velocity 3D field time series is an Eulerian field that represents a time series of three-dimensional velocity fields that are calculated from the frames in a particle 3D record. It can be created in a number of ways, using either a create velocity 3D field process (see chapter 32), one of the free processes that combine Eulerian fields (see chapter 22) or a velocity 3D field transform pipeline (see chapter 43).

A velocity 3D field time series includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described in detail later in this chapter.

Once created, a velocity 3D field time series object is displayed in the *Streams* object and Eulerian field list views.

13.2 Views

The velocity 3D field time series provides three views. These are

- details view
- calculator view
- process view

13.2.1 Details view

Description

The details view gives information about the velocity field. This information includes its name and description, details about the spatial grid on which the velocity field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the velocity field object.

13.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the velocity field. Currently 33 calculators are available and these are discussed in detail in section 13.4.

Access

Access to the calculator view is through the popup menu of the velocity field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators*… this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator*… this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator*… this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators*… this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

13.2.2.1 Custom calculator dialog

When the user chooses to create a custom calculator a dialog box displayed. In contains the following tabs:

Name

On this tab the user can specify the name of the calculator. If the user leaves this field blank the formula text is substituted for the name. This name is displayed in the calculator list in the calculator view.

Formula

For a scalar field calculator a single text area is provided for the user to enter the formula for the calculator. This formula is defined by the rules

described in chapter 49. For a vector field calculator three text areas are provided for entering the formulae for the x, y and z components of the field.

13.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the velocity field object.

13.3 Popup menu

The velocity field provides the following options in its popup menu.

- *Save as*… – saves the velocity field in a new file. All velocity field files have the extension *.vel3d*. The new file name is displayed next to the velocity field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the velocity field to disk. If the velocity field currently has no associated file then the user is prompted for a file name. All velocity field files have the extension .vel3d.
- *Save* fields to text file …– saves the velocity field to disk as a text file. The user is presented with a dialog box in which they can choose the file, the number of decimal places for the field values, and whether derivative fields should be stored as well. The resulting text file has the following comma delimited format. In the first line is the size of the field, the number of time steps, and the number of grid points in the x, y and z directions. Then for each time step data is presented in the following way. The time is output on the first line. Then for each value of z the z value is output on the next line. Then for each field that is output (this could be just u, v and w or include the 9 derivative fields, in order $\partial u/\partial x$, $\partial u/\partial y$, $\partial u/\partial z$, etc) a table is written to the file. The first row is the x coordinates of each grid point. In the following rows the first entry is the y coordinate followed by the field value at the corresponding grid point. The string NAN is output if the field is undefined.
- *Change name*… – allows the user to change the name of the velocity field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description*… – allows the user to change the description of the velocity field.

- *Delete fields…* – this option allows the user to delete fields within the velocity field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the velocity field on disk. To save space it may be convenient to delete all of the calculated fields before saving the velocity field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the velocity field from *Streams*. All views are closed. If the velocity field has unsaved data then the user is prompted to save this data before closure.

13.4 Calculators

Each velocity field contains a number of predefined calculators that allow the user to extract information of interest from the velocity field. Each calculator produces a four dimensional field derived from the original velocity field. The available calculators are shown in table 13.1.

Each of these calculators has a single view, the manager view. This view provides the user with control over which projection of the calculated field they wish to compute.

The manager view is common to all Eulerian 3D field calculators and is discussed in detail in section 13.4.1.

Table 13.1 The list of available velocity field calculators.

Symbol	Description
(u,v,w)	the instantaneous velocity vector
|(u,v,w)|	the magnitude of the instantaneous velocity vector
u	the x component of the instantaneous velocity vector
v	the y component of the instantaneous velocity vector
w	the z component of the instantaneous velocity vector
KE	the 3D kinetic energy based on the instantaneous velocity field.
div(u,v,w)	the 3D divergence of the instantaneous velocity field
ω	the vorticity vector field
$\|(\omega_x, \omega_y, \omega_z)\|$	the magnitude of the vorticity vector field
ω_x	the x component of the vorticity vector field
ω_y	the y component of the vorticity vector field
ω_z	the z component of the vorticity vector field
∂u/∂x	partial derivative of u with respect to x.
∂u/∂y	partial derivative of u with respect to y.
∂u/∂z	partial derivative of u with respect to z.
∂v/∂x	partial derivative of v with respect to x.
∂v/∂y	partial derivative of v with respect to y.
∂v/∂z	partial derivative of v with respect to z.
∂w/∂x	partial derivative of w with respect to x.
∂w/∂y	partial derivative of w with respect to y.
∂w/∂z	partial derivative of w with respect to z.
(u',v',w')	the fluctuating velocity vector. The time averaged velocity vector, calculated from the entire time series is subtracted from (u,v,w) to generate this vector.
tke	the 3D turbulent kinetic energy.
u'u'	the intensity of the fluctuating x velocity component.
v'v'	the intensity of the fluctuating y velocity component.
w'w'	the intensity of the fluctuating z velocity component.
-u'v'	the negative correlation between the fluctuating x and y velocity components.

-v'w'	the negative correlation between the fluctuating x and z velocity components.
-u'v'	the negative correlation between the fluctuating y and z velocity components.
u'	the x component of the fluctuating velocity field
v'	the y component of the fluctuating velocity field
w'	the z component of the fluctuating velocity field
Coverage	the coverage is defined to be 1 at a grid point at which the velocity is defined and 0 if it is not defined.

13.4.1 Manager View

Description

Each calculator for an Eulerian 3D field time series provides a single view known as the manager view. The manager view allows the user to calculate any one of the 81 different projections that the field calculator provides. All field calculators have the same manager view, and therefore this view will be described in a generic way.

Each field calculator generates a derived four-dimensional field from the velocity field object for which it is defined. The user is at liberty to extract a range of different projections of that four-dimensional field by specifying constraints on the four independent variables x, y, z and t. For each variable the user can select

- to leave that variable as a variable in the calculated field, specifying the range for that variable, or
- to average over that variable, specifying the range of that variable over which the average is to be calculated, or
- to choose a value for that variable, at which the field will be calculated.

A set of radio buttons, text boxes and drop down boxes allow the user to select the projection of interest. Once these controls have been set the field is calculated when the Calculate field button is pressed. As the calculation of some fields can take some seconds the computation is run in a separate thread so that the user can perform other tasks if required. Progress of the calculation is provided in a monitor dialog. Once calculated the field is displayed in the list box labelled Calculated fields. Each field is given a unique name and an icon that indicates the type of field it represents (scalar, or vector, 0, 1, 2, 3 or 4 dimensional).

The fields are given two names that indicate their genesis. The first, short name, matches the name of the calculator itself. The name of the second, longer name, begins with the field calculator name but this is followed by parentheses containing three groups of symbols, separated by commas, referring to the variables, x, y, z and t respectively (except in the case of a cut). These symbols are

- the variable name, if that variable was chosen to be variable, followed by its range, or
- the variable name in angle brackets (<>) if that variable was averaged, followed by the limits for the averaging process, or
- the variable name, followed by its value, if that variable was fixed.

Finally the name is completed with the name of the velocity 3D field time series preceded by the word "from".

One or more fields can be selected and deleted by clicking the Delete field button, or they can be closed using the field object's popup menu.

The Calculated fields list supports the drag action so that fields in this list can be dragged (by clicking and holding the left mouse button and moving the mouse) to a list box that supports the drop action for fields. An example is the list of fields in the analysis view of a tablet (see the *User's Guide: Field Objects* manual chapter 12).

When a velocity field object is saved to disk all of its calculators and their calculated fields are saved with it. As these fields can be easily recalculated, if storage space is at a premium, the user may wish to delete all fields before saving the velocity field object. This is particularly true of three-dimensional velocity fields, which can contain very large volumes of data.

Access

The manager view is opened by double clicking on the field calculator in the intensity field calculator view, or through the popup menu of the field calculator object.

Popup Menu

When a field object in the Calculated field list is right clicked its popup menu is displayed, and when it is double clicked its default view is opened. If multiple field objects are selected a right click of the mouse will display a separate popup menu with the following options:

- *Open default views* – this option will open the default view for each of the selected field objects.
- *Close all views* – this option will close all of the views of the selected field objects.

13.4.2 Custom calculators

Velocity fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49). See section 11.2.2 for more details.

13.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 4 dimensional field in a comma (,) delimited text file. Each three-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

14 Velocity-Intensity Field Time Series

14.1 Description

A velocity-intensity field time series is the combination of a velocity 2D field time series and an intensity field time series where it is assumed that these two fields are associated with the velocity and a scalar field (e.g. concentration or density) of the same fluid flow, captured, for example, through a PTV-LIF laboratory system. The velocity and intensity fields are created in the ways described in previous chapters and the dual field is created by simply combining two previously created fields. These fields must be compatible – a requirement that both share the same grid structure. In practice this means that the two time series contain the same number of 2D fields with the same time step between them, and that the two fields within the time series have the same physical grid (same number of grid points in each direction and the same spacing between grid points). This grid consistency is most easily achieved by using the same grid in the interpolation processes whereby the fields were originally created. The initial time of the two time series need not match, nor the origin of the spatial grids. Imagine the velocity and intensity fields being captured by two separate cameras. These cameras might start at different times and their images might be slightly offset with respect to one another. Neither of these differences is important for the combined field. It should be noted that if the time and space origins do differ then the origins defined in the velocity field replace those in the intensity field to ensure consistency between the combined fields.

14.2 Views

The velocity-intensity field time series provides three views. These are

- details view
- calculator view
- process view

14.2.1 Details view

Description

The details view gives information about the velocity-intensity field. This information includes its name and description, details about the spatial grid on which the velocity-intensity field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the velocity-intensity field object.

14.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the velocity-intensity field. Currently 43 calculators are available and these are discussed in detail in section 14.4.

Access

Access to the calculator view is through the popup menu of the velocity-intensity field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

14.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the velocity-intensity field object.

14.3 Popup menu

The velocity-intensity field provides the following options in its popup menu.

- *Save as…* – saves the velocity-intensity field in a new file. All velocity – intensity field files have the extension .vin. The new file name is displayed next to the velocity-intensity field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the velocity-intensity field to disk. If the velocity-intensity field currently has no associated file then the user is prompted for a file name. All velocity field files have the extension .vin.
- *Change name…* – allows the user to change the name of the velocity-intensity field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description…* – allows the user to change the description of the velocity-intensity field.
- *Delete fields…* – this option allows the user to delete fields within the velocity-intensity field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the velocity field on disk. To save space it may be convenient to delete all of the calculated fields before saving the velocity-intensity field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.

- *Close* – removes the velocity-intensity field from *Streams*. All views are closed. If the velocity-intensity field has unsaved data then the user is prompted to save this data before closure.

14.4 Calculators

Each velocity-intensity field contains a number of predefined calculators that allow the user to extract information of interest from the velocity-intensity field. Each calculator produces a three dimensional field derived from the original velocity-intensity field.

The available calculators are shown in table 14.1. All but the first five of these calculators operate on the velocity or intensity fields only. The first five operate on the combined fields.

Each of these calculators has a single view, the manager view. This view provides the user with control over which projection of the calculated field they wish to compute.

The manager view is common to all Eulerian 2D field calculators and is discussed in detail in section 11.4.1.

Table 14.1 The list of available velocity-intensity field calculators.

Symbol	Description
(uI,vI)	the instantaneous scalar flux vector
uI	the x component of the instantaneous scalar flux vector
vI	the y component of the instantaneous scalar flux vector
u'I'	the product of the x component of the fluctuating velocity and the fluctuating intensity.
v'I'	the product of the y component of the fluctuating velocity and the fluctuating intensity.
(u,v)	the instantaneous velocity vector
(u',v')	the fluctuating velocity vector. The time averaged velocity vector, calculated from the entire time series is subtracted from (u,v) to generate this vector.
|uv|	the magnitude of the instantaneous velocity vector
u	the x component of the instantaneous velocity vector

v	the y component of the instantaneous velocity vector
($\partial u/\partial t$, $\partial v/\partial t$)	the local acceleration vector
coverage	the coverage is defined to be 1 at a grid point at which the velocity is defined and 0 if it is not defined.
KE	the 2D kinetic energy based on the instantaneous velocity field.
ω	the vorticity field
div(u,v)	the divergence of the instantaneous velocity field
strain rate	the strain rate of the instantaneous velocity field
TKE	the 2D turbulent kinetic energy. The time average of this field gives the 2D TKE per unit mass.
-u'v'	the negative correlation between the fluctuating velocity components. The time average of this field gives the turbulent shear stress per unit mass
u'u'	the intensity of the fluctuating x velocity component.
v'v'	the intensity of the fluctuating y velocity component.
u'u'u'	triple velocity fluctuation correlation.
u'u'v'	triple velocity fluctuation correlation.
u'v'v'	triple velocity fluctuation correlation.
v'v'v'	triple velocity fluctuation correlation.
u'	the x component of the fluctuating velocity field
v'	the y component of the fluctuating velocity field
$\partial u/\partial x$	partial derivative of u with respect to x.
$\partial u/\partial y$	partial derivative of u with respect to y.
$\partial v/\partial x$	partial derivative of v with respect to x.
$\partial v/\partial y$	partial derivative of v with respect to y.
uT	the barotropic velocity. This is the depth (vertically) averaged horizontal velocity component. It is uniform in the vertical.
uC	the baroclinic velocity. The difference between the instantaneous horizontal velocity and the barotropic velocity.
(uT,0)	the barotropic velocity in vector form with a zero vertical velocity.
(uC,0)	the baroclinic velocity in vector form with a zero

	vertical velocity.
(uC,uT)	a vector whose two components are the baroclinic and barotropic velocity components.
I	the instantaneous intensity field
I'	the fluctuating intensity field. The time average intensity, calculated from the entire time series, is subtracted from I to generate this field.
I'I'	the intensity of the fluctuating intensity.
grad(I)	the gradient of the intensity field
grad(I)|	the magnitude of the gradient of the intensity field
∂I/∂x	partial derivative of intensity with respect to x
∂I/∂y	partial derivative of intensity with respect to y
∂I/∂t	partial derivative of intensity with respect to t

14.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

14.4.2 Custom calculators

Velocity-intensity fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49).

14.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

14.5 Creating velocity-intensity field time series

Velocity-intensity field time series are created by selecting Create velocity-intensity field time series from the *Streams* Eulerian menu. A dialog box is displayed that prompts the user for key information. Two tabs are displayed.

Name

- *Name* – the velocity-intensity field time series name, used to identify the time series from all others in the *Streams* system.

Field

In this tab the user chooses the velocity and intensity fields that will be combined to create the velocity-intensity field. All currently open velocity fields are listed in the *Possible velocity fields* list box and all currently open intensity fields in the *Possible intensity fields* list box. When the user selects a field in either of these lists *Streams* checks the fields in the other list and filters them so that only those that are compatible with the selected field are displayed. Once the user has selected a field in each list (because of the filter process only compatible fields can be selected) clicking the OK button will create the new velocity-intensity field time series and display it in the Object list and Eulerian list views. In order to save memory the two original fields are closed and removed from the system at this point.

At the bottom of this tab there is a single button. It performs the following task.

- *Restore* – resets both of the list boxes so that all currently loaded fields are displayed.

15 Density Field Time Series

15.1 Description

A density field time series is another Eulerian field that can be calculated from a particle 2D record. The density refers to the density of particles in each frame, not to the density of the fluid. While this quantity is not of interest in most PTV applications there may be circumstances where it is useful. For example the particles in the flow may be able to be used as tracers and tracer fluxes may be calculated from the change in particle distributions within the field of view.

A density field time series represents a time series of two-dimensional particle density fields that are calculated from the frames in a particle record. It can be created in a number of ways, using either a create density field process (see chapter 33), one of the free processes that combine Eulerian fields (see chapter 22) or a density field transform pipeline (see chapter 43).

A density field time series includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described later in this chapter.

Once created, a density field time series object is displayed in the *Streams* object and Eulerian field list views.

15.2 Views

The density field provides three views. These are

- details view
- calculator view
- process view

15.2.1 Details view

Description

The details view gives information about the density field. This information includes its name and description, details about the spatial

grid on which the density field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the density field object.

15.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the density field. Currently seven calculators are available and these are discussed in detail in section 15.4.

Access

Access to the calculator view is through the popup menu of the density field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

15.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the density field object.

15.3 Popup menu

The density field provides the following options in its popup menu.

- *Save as…* – saves the density field in a new file. All density field files have the extension *.den*. The new file name is displayed next to the density field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the density field to disk. If the density field currently has no associated file then the user is prompted for a file name. All density field files have the extension *.den*.
- *Change name…* – allows the user to change the name of the density field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description…* – allows the user to change the description of the density field.
- *Delete fields…* – this option allows the user to delete fields within the density field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the density field on disk. To save space it may be convenient to delete all of the calculated fields before saving the density field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the density field from *Streams*. All views are closed. If the density field has unsaved data then the user is prompted to save this data before closure.

15.4 Calculators

Each density field contains a number of predefined calculators that allow the user to extract information of interest from the density field. Each calculator produces a three dimensional field derived from the original density field.

The available calculators are shown in table 15.1.

Table 15.1 The list of available density field calculators.

Symbol	Description
ρ	the instantaneous density field
ρ'	the fluctuating density field. The time averaged density, calculated from the entire time series, is subtracted from rho to generate this field.
ρ' ρ'	the intensity of the fluctuating density.
grad(ρ)	the gradient of the density field
|grad(ρ)|	the magnitude of the gradient of the density field
∂ρ/∂x	partial derivative of density with respect to x
∂ρ/∂y	partial derivative of density with respect to y
∂ρ/∂t	partial derivative of the density with respect to t

15.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

15.4.2 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file*… – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

16 Shape Field Time Series

16.1 Description

A shape field represents a time series of two-dimensional shape fields that are calculated from the particle information stored in a particle 2D record. The term shape refers to a particle characteristic associated with the particle's geometry. Thus, wherever the word shape is used the terms effective diameter, area or aspect ratio could be substituted.

A shape field is created using a create shape field process (see chapter 34), one of the free processes that combine Eulerian fields (see chapter 22) or a shape field transform pipeline (see chapter 43).

The result is a three-dimensional scalar field (x, y and t are the three dimensions) where the chosen shape parameter is calculated on the nodes of a rectangular grid.

A shape field includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described later in this chapter.

Once created, a shape field object is displayed in the *Streams* object and Eulerian field list views.

16.2 Views

The shape field provides three views. These are

- details view
- calculator view
- process view

16.2.1 Details view

Description

The details view gives information about the shape field. This information includes its name and description, details about the spatial grid on which the shape field is defined, the time step between fields and

the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the shape field object.

16.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the shape field. Currently seven calculators are available and these are discussed in detail in section 16.4.

Access

Access to the calculator view is through the popup menu of the shape field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Delete selected calculators…* this option deletes all of the selected calculators from the shape field object. The user is prompted to confirm the action.

16.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the shape field object.

16.3 Popup menu

The shape field provides the following options in its popup menu.

- *Save as…* – saves the shape field in a new file. All shape field files have the extension *.shp*. The new file name is displayed next to the shape field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the shape field to disk. If the shape field currently has no associated file then the user is prompted for a file name. All shape field files have the extension *.shp*.
- *Change name…* – allows the user to change the name of the shape field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description…* – allows the user to change the description of the shape field.
- *Delete fields…* – this option allows the user to delete fields within the shape field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the shape field on disk. To save space it may be convenient to delete all of the calculated fields before saving the shape field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the shape field from *Streams*. All views are closed. If the shape field has unsaved data then the user is prompted to save this data before closure.

16.4 Calculators

Each shape field contains a number of predefined calculators that allow the user to extract information of interest from the shape field. Each calculator produces a three dimensional field derived from the original shape field.

The available calculators are shown in table 16.1.

Table 16.1 The list of available shape field calculators. S is a placeholder and is replaced by D, A and AR for effective diameter, area and aspect ratio field time series respectively.

Symbol	Description
S	the instantaneous shape field.
S'	the fluctuating shape field. The time averaged shape field, calculated from the entire time series, is subtracted from S to generate this field.
S'S'	the intensity of the fluctuating shape field.
grad(S)	the gradient of the shape field. This is a vector field.
|grad(S)|	the magnitude of the gradient of the shape field.
dS/dx	partial derivative of the shape field with respect to x.
dS/dy	partial derivative of the shape field with respect to y.
dS/dt	the time derivative of the shape field.

16.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

16.4.2 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

17 Displacement 2D Field Time Series

17.1 Description

A displacement 2D field time series is an Eulerian field that represents a time series of two-dimensional displacement fields that are calculated from the particle information stored in a particle 2D record. Displacement is simply the change in position of each particle between frames. Displacement fields are useful in analyzing the behavior of solid materials under loading.

A displacement field is created using a create displacement 2D field process (see chapter 35), or a displacement 2D field transform pipeline (see chapter 43).

The result is a three-dimensional scalar field (x, y and t are the three dimensions) where the material displacement is calculated on the nodes of a rectangular grid. As described in chapter 35 the grid on which the displacement is computed may either be fixed in space or fixed to the material.

A displacement field includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described later in this chapter.

Once created, a displacement field object is displayed in the *Streams* object and Eulerian field list views.

17.2 Views

The displacement 2D field time series provides three views. These are

- details view
- calculator view
- process view

17.2.1 Details view

Description

The details view gives information about the displacement field. This information includes its name and description, details about the spatial grid on which the shape field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the displacement field object.

17.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the displacement field. Currently thirteen calculators are available and these are discussed in detail in section 17.4.

Access

Access to the calculator view is through the popup menu of the displacement field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the displacement field object. The user is prompted to confirm the action.

17.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the displacement field object.

17.3 Popup menu

The displacement 2D field time series provides the following options in its popup menu.

- *Save as…* – saves the displacement field in a new file. All displacement field files have the extension *.dis2d*. The new file name is displayed next to the displacement field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the displacement field to disk. If the displacement field currently has no associated file then the user is prompted for a file name. All displacement field files have the extension *.dis2d*.
- *Change name…* – allows the user to change the name of the displacement field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description…* – allows the user to change the description of the displacement field.
- *Delete fields…* – this option allows the user to delete fields within the displacement field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the displacement field on disk. To save space it may be convenient to delete all of the calculated fields before saving the displacement field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.

- *Close* – removes the displacement field from *Streams*. All views are closed. If the displacement field has unsaved data then the user is prompted to save this data before closure.

17.4 Calculators

Each displacement 2D field times series contains a number of predefined calculators that allow the user to extract information of interest from the displacement field. Each calculator produces a three dimensional field derived from the original displacement field.

The available calculators are listed in table 17.1.

Table 17.1 The list of available displacement field calculators.

Symbol	Description
(Xi,Yi)	the incremental displacement vector – i.e. the displacement between frames.
|XiYi|	the magnitude of the incremental displacement vector.
Xi	the x component of the incremental displacement vector.
Yi	the y component of the incremental displacement vector.
(Xc,Yc)	the cumulative displacement vector. This corresponds to the sum of all displacements at a given grid point from the first frame.
|XcYc|	the magnitude of the cumulative displacement vector.
Xc	the x component of the cumulative displacement vector.
Yc	the y component of the cumulative displacement vector.
ε xx	the direct x cumulative strain field.
ε yy	the direct y cumulative strain field.
ε xy	the cumulative shear strain.
ε vol	the cumulative volumetric strain.
rot	the rotation.

17.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

17.4.2 Custom calculators

Displacement fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49).

17.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

18 Displacement 3D Field Time Series

18.1 Description

A displacement 3D field time series is an Eulerian field that represents a time series of three-dimensional displacement fields that are calculated from the particle information stored in a particle 3D record. Displacement is simply the change in position of each particle between frames. Displacement fields are useful in analyzing the behavior of solid materials under loading.

A displacement field is created using a create displacement 3D field process (see chapter 36), or a displacement 3D field transform pipeline (see chapter 43).

The result is a four-dimensional scalar field (x, y, z and t are the three dimensions) where the material displacement is calculated on the nodes of a rectangular grid. As discussed in chapter 36 the user may choose the grid to be fixed in space or fixed to the material being observed.

A displacement field includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described later in this chapter.

Once created, a displacement field object is displayed in the *Streams* object and Eulerian field list views.

18.2 Views

The displacement 3D field time series provides three views. These are

- details view
- calculator view
- process view

18.2.1 Details view

Description

The details view gives information about the displacement field. This information includes its name and description, details about the spatial grid on which the shape field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the displacement field object.

18.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the displacement field. Currently twenty calculators are available and these are discussed in detail in section 18.4.

Access

Access to the calculator view is through the popup menu of the displacement field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the displacement field object. The user is prompted to confirm the action.

18.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the displacement field object.

18.3 Popup menu

The displacement 3D field time series provides the following options in its popup menu.

- *Save as…* – saves the displacement field in a new file. All displacement field files have the extension *.dis3d*. The new file name is displayed next to the displacement field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the displacement field to disk. If the displacement field currently has no associated file then the user is prompted for a file name. All displacement field files have the extension *.dis3d*.
- *Change name…* – allows the user to change the name of the displacement field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description…* – allows the user to change the description of the displacement field.
- *Delete fields…* – this option allows the user to delete fields within the displacement field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields…* – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the displacement field on disk. To save space it may be convenient to delete all of the calculated fields before saving the displacement field to disk.
- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.

- *Close* – removes the displacement field from *Streams*. All views are closed. If the displacement field has unsaved data then the user is prompted to save this data before closure.

18.4 Calculators

Each displacement 3D field times series contains a number of predefined calculators that allow the user to extract information of interest from the displacement field. Each calculator produces a three dimensional field derived from the original displacement field.

The available calculators are shown in table 18.1.

Table 18.1 The list of available displacement field calculators.

Symbol	Description
(Xi,Yi,Zi)	the incremental displacement vector – i.e. the displacement between frames.
|XiYiZi|	the magnitude of the incremental displacement vector.
Xi	the x component of the incremental displacement vector.
Yi	the y component of the incremental displacement vector.
Zi	the z component of the incremental displacement vector.
(Xc,Yc,Zc)	the cumulative displacement vector. This corresponds to the sum of all displacements at a given grid point from the first frame.
|XcYcZc|	the magnitude of the cumulative displacement vector.
Xc	the x component of the cumulative displacement vector.
Yc	the y component of the cumulative displacement vector.
Zc	thez component of the cumulative displacement vector.
ε xx	the direct x cumulative strain field.
ε yy	the direct y cumulative strain field.

Symbol	Description
ε zz	the direct z cumulative strain field.
ε xy	the cumulative xy shear strain field.
ε xz	the cumulative xz shear strain field.
ε yz	the cumulative yz shear strain field.
ε vol	the cumulative volumetric strain.
rot x	the rotation about the x axis.
rot y	the rotation about the y axis.
rot z	the rotation about the z axis.

18.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

18.4.2 Custom calculators

Displacement fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49).

18.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 4 dimensional field in a comma (,) delimited text file. Each three-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

19 Displacement 3D Surface Field Time Series

19.1 Description

A displacement 3D surface field time series is an Eulerian field that represents a time series of displacement fields associated with a surface in three dimensional space – for example a material surface that deforms over time. It is calculated from the particle information stored in a particle 3D record. In constructing the surface field it is assumed that the surface can be projected through a one to one mapping onto the x-y plane. Thus complex surfaces that bend back onto themselves cannot be represented. It is also important that the user chooses a three dimensional coordinate system for their particle 3D record that is consistent with this convention.

A displacement field is created using a create displacement 3D surface field process (see chapter 37), or a displacement 3D surface field transform pipeline (see chapter 43).

The result is a three-dimensional scalar field (x, y and t are the three dimensions) where the material displacement is calculated on the nodes of a rectangular grid. In order for the 3D surface to be able to be reconstructed the elevation (z coordinate) of the surface at each grid point is stored.

A displacement field includes a number of calculators that allow the user to explore the data contained within the time series. These calculators are described later in this chapter.

Once created, a displacement field object is displayed in the *Streams* object and Eulerian field list views.

19.2 Views

The displacement 3D surface field time series provides three views. These are

- details view
- calculator view

- process view

19.2.1 Details view

Description

The details view gives information about the surface displacement field. This information includes its name and description, details about the spatial grid on which the shape field is defined, the time step between fields and the total number of fields in the time series. This data cannot be changed in this view.

Access

Access to the details view is through the popup menu of the displacement field object.

19.2.2 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the displacement field. Currently seventeen calculators are available and these are discussed in detail in section 19.4.

Access

Access to the calculator view is through the popup menu of the displacement field object or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).

- *Delete selected calculators*… this option deletes all of the selected calculators from the displacement field object. The user is prompted to confirm the action.

19.2.3 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the displacement field object.

19.3 Popup menu

The displacement 3D surface field time series provides the following options in its popup menu.

- *Save as*… – saves the displacement field in a new file. All surface displacement field files have the extension *.dis3dsur*. The new file name is displayed next to the displacement field name in the *Streams* object and Eulerian field list views.
- *Save* – saves the displacement field to disk. If the displacement field currently has no associated file then the user is prompted for a file name. All displacement field files have the extension . *dis3dsur*.
- *Change name*… – allows the user to change the name of the displacement field. This name change will be reflected in the *Streams* object list and Eulerian list views.
- *Edit description*… – allows the user to change the description of the displacement field.
- *Delete fields*… – this option allows the user to delete fields within the displacement field. Deletion must occur from the beginning or end of the time series, never from the middle. This action requires confirmation from the user.
- *Open details view* – opens the details view.
- *Open calculator view* – opens the calculator view. This is the default view.
- *Open process view* – opens the process view.
- *Delete calculated fields*… – deletes all of the calculated fields that currently exist within the field calculators. The calculators and their calculated fields are all stored with the displacement field on disk. To save space it may be convenient to delete all of the calculated fields before saving the displacement field to disk.

- *Execute process pipeline* – executes the process pipeline for the field object provided it is valid.
- *Close all views* – closes all open views for this object.
- *Close* – removes the displacement field from *Streams*. All views are closed. If the displacement field has unsaved data then the user is prompted to save this data before closure.

19.4 Calculators

Each displacement 3D surface field times series contains a number of predefined calculators that allow the user to extract information of interest from the displacement field. Each calculator produces a three dimensional field derived from the original displacement field.

The available calculators are shown in table 19.1.

Table 19.1 The list of available displacement field calculators.

Symbol	Description
z	the surface elevation relative to the x-y plane.
$\partial z/\partial x$	the gradient of the surface in the x direction.
$\partial z/\partial y$	the gradient of the surface in the y direction.
(Xi,Yi,Zi)	the incremental displacement vector – i.e. the displacement between frames.
|XiYiZi|	the magnitude of the incremental displacement vector.
Xi	the x component of the incremental displacement vector.
Yi	the y component of the incremental displacement vector.
Zi	the z component of the incremental displacement vector.
(Xc,Yc,Zc)	the cumulative displacement vector. This corresponds to the sum of all displacements at a given grid point from the first frame.
|XcYcZc|	the magnitude of the cumulative displacement vector.
Xc	the x component of the cumulative

Symbol	Description
	displacement vector.
Yc	the y component of the cumulative displacement vector.
Zc	the z component of the cumulative displacement vector.
eps xx	the direct x cumulative strain field.
eps yy	the direct y cumulative strain field.
eps xy	the cumulative shear strain.
rot z	the rotation about the z axis.

19.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. This is discussed in detail in section 11.4.1.

19.4.2 Custom calculators

Displacement fields also provide the capability to define custom calculators. Both scalar and vector field producing calculators can be defined. The only difference between the two is that the first requires a single mathematical formula while the second requires two (see chapter 49).

19.4.3 Popup menu

A field calculator provides the following options in its popup menu.

- *Save to text file…* – saves the full 3 dimensional field in a comma (,) delimited text file. Each two-dimensional field in the time series is stored in sequence.
- *Open manager view* – opens the manager view.

20 Lagrangian 2D Path Fields

20.1 Description

Lagrangian 2D path fields store particle path information derived from a particle 2D record. Particle matches must have been created for the record through the application of a PTV 2D analysis pipeline. A Lagrangian path field contrasts with the Eulerian velocity field described in chapter 12 in that flow information (actually particle information) is particle-based instead of being defined on a spatial grid.

Particle location, velocity, radius and intensity are automatically stored at each location along the particle path and the user is at liberty to request that additional shape information, aspect ratio, theta and eccentricity are also stored. The availability of this additional information will depend on whether this data was generated during the particle identification process.

The data stored in a Lagrangian 2D path field can be thought of in two ways. Firstly as a set of particle paths, each of which may start at a different time, and end at a different time. Secondly as a two-dimensional matrix of data where the rows (or columns) are labeled by the path number and the columns (or rows) are labeled by time. This rectangular matrix will contain elements that are undefined because a particular path may not exist at that time. The paths are ordered by the time at which they start. Thus the earlier they begin the earlier they appear in the set of paths. However, there is no order imposed on paths that start at the same time.

Once created, a Lagrangian 2D path field object is displayed in the *Streams* object and Lagrangian field list views.

20.2 Views

The Lagrangian path field provides seven views. These are

- calculator view
- details view
- particles view
- path view
- path list view

- process view
- selected path list view

20.2.1 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the Lagrangian field. Currently 20 calculators are available and these are discussed in detail later in this chapter.

Access

Access to the calculator view is through the popup menu of the Lagrangian field object, or by double clicking the object

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

20.2.1.1 Custom calculator dialog

When the user chooses to create a custom calculator a dialog box displayed. In contains the following tabs:

Name

On this tab the user can specify the name of the calculator. If the user leaves this field blank the formula text is substituted for the name. This name is displayed in the calculator list in the calculator view.

Formula

For a scalar field calculator a single text area is provided for the user to enter the formula for the calculator. This formula is defined by the rules described in chapter 49. For a vector field calculator two text areas are provided for entering the formulae for the x and y components of the field.

20.2.2 Details view

Description

The details view gives information about the Lagrangian path field. This information includes its name and description, details about the start and end time for the field, the number of paths, and the extent of the physical domain. Only the first two of these, along with the spatial and temporal units, can be changed within this view. The Save and Cancel buttons enable the user to save or discard changes made in this view.

Access

Access to the details view is through the popup menu of the Lagrangian field object.

20.2.3 Particles view

Description

The particles view displays information about a specific path within the Lagrangian field. All of the particle locations along the path are presented in tabular form. For each particle within the path the table lists its number, the time, its position, the two components of its velocity, its radius, intensity and area, and, if available, its aspect ratio, theta and eccentricity.

Access

Access to the particles view is through the popup menu of a path object. Paths are displayed in the path list view and the selected path list view of the Lagrangian field object.

20.2.4 Path view

Description

The path view has much in common with the particle 2D record particle view in that it displays particles and provides a simply video interface to observe the particle movement. However, the focus in this view is on paths. The user can select one or more frames to display starting at a particular time. All particles present in the selected frames are displayed. The user may choose to display the paths associated with the particles present in the first frame displayed. The backward paths button displays the portion of the path before the first frame in yellow. The core paths button displays the path encapsulated by the frames currently displayed in red. And the forward paths button displays the portion of the path that exists beyond the last frame in blue. As the animation moves forward in time paths will come and go and the backward, core and forward portions will change. Path numbers can be displayed if desired.

Particles on each path can be displayed as points, or in their true size, shape and orientation. In addition the particles can be drawn in their true colour or in monochrome, and the particle velocities can be indicated with vectors. Thus most of the particle data are available to the user.

The view has four toolbars displayed above the path display area. These toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls that allow the user to control the animation features of the view.

- *Current time* – this combo box displays the first frame currently displayed as well as its time. The user can select any value in this box to move to a different time in the field.
- *Number of frames displayed* – this combo box selects the number of frames (and corresponding time) overlaid in the view.
- *Go to next time* – steps the display forward one frame.
- *Go to previous time*– steps the display back one frame.
- *Play*– starts the animation forward in time.
- *Play in reverse* – starts the animation backward in time.
- *Pause* – pauses the animation.
- *Go to start* – resets the display to the first frame.
- *Go to end* – resets the display to the last frame.

- *Animation timing delay (ms)* – this text box allows the user to change the speed of the animation. The displayed timing delay is the time step for the record. The actual delay may be longer if the drawing overhead is high.

Path toolbar

The path tool bar sits below the animation toolbar. It allows the user to select the paths, particles and vectors that are displayed in the view.

- *Particles* – when this button is pressed all the particles in the visible paths are displayed.
- *Forward paths* – displays the forward paths
- *Core paths* – displays the core paths
- *Backward paths* – displays the backward paths
- *Vectors* – displays the velocity vectors at each point on each path.
- *Numbers* – displays the path numbers at the start of each path.

Cursor toolbar

The cursor toolbar, second from the bottom, displays the location of the cursor on screen and indicates the current drawing tool.

- *x* – This text box displays the x coordinate (in the specified spatial units) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *y* – This text box displays the y coordinate (in the specified spatial units) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *Drawing tool* – This text box displays the name of the currently selected drawing tool. This tool is used for defining regions in the view. The tool can be changed through the *Region* menu.

View toolbar

The view toolbar sits at the bottom of the toolbar area. Here the user can select the portion of the frame that they wish to display. The zoom capability, described below, provides an alternative technique for changing the region of the frame that is displayed.

- *Min x axis* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn to the correct scale – thus units in the x and y directions have the same length on screen.

- Max x axis – This text box displays the maximum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Min y axis – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.
- Max y axis – This text box displays the maximum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible.

Menus

File Menu

- *Save as image*… – allows the user to save the path view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user has considerable control over the stored image including choice of font size, line thickness, resolution etc, and is able to preview the image before it is stored on disk. Further details of this process are provided in the User's Guide: Field Objects.
- *Save as image sequence*… allows the user to save each of the frames in the path field as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the User's Guide: Field Objects.

View Menu

- *Set layout*… – – this displays a dialog box that enables the user to set a range of display parameters common to all Lagrangian path field path views (i.e. changing these settings for this view changes them for all path views). Details regarding this dialog box are provided in section 20.2.4.1).
- *Draw point particles*… – a check menu item that determines whether the particles are displayed as points.
- *Draw true particles*… – a check menu item that determines whether the particles are displayed as their true shapes and sizes.
- *Draw monochrome particles*… – a check menu item that determines whether the particles are drawn in white.
- *Draw true colour particles*… – a check menu item that determines whether the particles are drawn using their true colour.
- *Show particles*… – a check menu item that determines whether the particles are displayed. This has the same effect as the equivalent button in the paths toolbar.

- *Show forward paths* – a check menu item that determines whether the forward paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show core paths* – a check menu item that determines whether the core paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show backward paths* – a check menu item that determines whether the backward paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show vectors* – a check menu item that determines whether the vectors paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show path numbers* – displays the path numbers at the start of each path.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y and t variables can be set. Sensible minimum values are set for these.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar.
- Adjust view size – this resizes the view window to minimize waste space. The same effect can be achieved by double clicking in the blank space on either side of the toolbars (assuming that they are visible).

Animation Menu

- *Go to next time*… – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to previous time*…– steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Play*…– starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Region Menu

- *Manage regions*… – displays all of the currently defined regions in a list box. Double clicking a region in this list box enables the user to edit its parameters.

- *Load regions…*– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn* or*.rgn2* extension.
- *Save regions as…*– saves all of the currently defined regions in a *.rgn2* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.
- *Draw internal rectangle* – sets the current drawing tool to an internal rectangle. This is the default.
- *Draw internal circle* – sets the current drawing tool to an internal circle.
- *Draw internal polygon* – sets the current drawing tool to an internal polygon.
- *Draw internal freehand* – sets the current drawing tool to an internal freehand.
- *Draw external rectangle* – sets the current drawing tool to an external rectangle.
- *Draw external circle* – sets the current drawing tool to an external circle.
- *Draw external polygon* – sets the current drawing tool to an external polygon.
- *Draw external freehand* – sets the current drawing tool to an external freehand.

Zoom Menu

- *Zoom…* – causes the view to be zoomed. If only one internal rectangle is currently defined then it is used as the zoom rectangle. If more than one internal rectangle exists then a dialog box is presented to the user, listing all internal rectangles currently defined. The user can select a zoom rectangle from the list.
- *Unzoom…*– this undoes any zoom operation by redisplaying the entire video frame area.

The path view provides the user with two different popup menus. The first is displayed if the mouse is right clicked within a region drawn on the view. The second is displayed if the mouse is right clicked outside any regions.

Popup menu 1

- *Delete paths in region…* – deletes all paths that have a particle location within the specified region for the frames currently displayed. If a path passes through the region without "touching down" then the path is not deleted.
- *Open path list view…* – displays the path list view for the paths that have a particle location in the region for the frames currently displayed.
- *Zoom…* – if the mouse is within an internal rectangle this region is used as a zoom rectangle. This means that this region defines the displayed area. It is effectively a shorthand way of changing the text fields in the view toolbar. Complications arise if the mouse lies within more than one rectangular region, or within a region that is not an internal rectangle. In the first case the most recently defined rectangle is used as the zoom rectangle, while in the second case a dialog box is presented to the user, listing all internal rectangles currently defined. The user can select a zoom rectangle from the list.
- *Unzoom…* – this undoes any zoom operation by redisplaying the entire video frame.

Popup Menu 2

- *Set layout…* – see View menu.
- *Go to next time…* – see Animation menu.
- *Go to previous time…*– see Animation menu.
- *Play…*– see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show particles…* – see View menu.
- *Show forward paths* – see View menu.
- *Show core paths* –see View menu.
- *Show backward paths* – see View menu.
- *Show vectors* – see View menu.
- *Show path numbers* – see View menu.
- *Zoom…* see Zoom menu.
- *Unzoom…*– see Zoom menu.

Access

Access to the path view is through the popup menu of the Lagrangian field.

20.2.4.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all path views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Vector scale factor* – allows the user to change the length of velocity vectors.
- *Line width* – the line width for the view is specified here. The default is 1.5.
- *Font size* – sets the font size. The default is 14.
- *Rotate y label* – when selected the y axis label is rotated so that the text is vertical.
- *Show info label* – if selected the text at the bottom of the view giving the time of the frames displayed is visible.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Points colour* – the colour of point particles. It can be changed by clicking on the Select… button.

- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *Forward paths colour* – the colour of lines for forward paths. It can be changed by clicking on the Select… button.
- *Core paths colour* – the colour of lines for core paths. It can be changed by clicking on the Select… button.
- *Backward paths colour* – the colour of lines for backward paths. It can be changed by clicking on the Select… button.
- *Velocity vector colour* – the colour used for drawing velocity vectors. It can be changed by clicking on the Select… button.
- *Background colour* – The background colour. It can be changed by clicking on the Select… button.

20.2.5 Path list view

Description

The path list view lists all of the paths in the field in tabular form. The table includes the following data associated with a path: its number, the frame in which it starts, its length (in frames), the start and end points, the physical length of the path or the distance travelled by the particle, the total displacement from the start point to the end point (a vector), the mean velocity of the particle along the path, the mean radius of the particle along its path, and the standard deviations of the velocity and radius along its path.

Details of particular paths can be obtained by right clicking on a path in the list and choosing to open the particles view.

Menus

File Menu

- *Save to text file…* – saves the paths (a subset of which can be selected by the user) to a text file with a .txt or .csv extension. The text file provides the following information for each path line – number, start frame, length, x and y coordinates of the start and end point of the path, total distance travelled, net displacement in the x and y directions, the average u and v velocities, the average radius of the particle along the path, and the standard deviations of u, v, and radius along the path.

View Menu

- *Change decimal places…* – changes the number of decimal places used to display the parameters in the table.

Sort Menu

- *Sort by number* – sorts the path table by the path number.
- *Sort by start x* – sorts the path table by the initial x position.
- *Sort by start y* – sorts the path table by the initial y position.
- *Sort by end x* – sorts the path table by the last x position.
- *Sort by end y* – sorts the path table by the last y position.
- *Sort by x disp* – sorts the path table by the x displacement along the path.
- *Sort by y disp* – sorts the path table by the y displacement along the path.

Access

Access to the path list view is through the popup menu of the Lagrangian path field object.

20.2.6 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the Lagrangian path field object.

20.2.7 Selected path list view

Description

The selected path list view lists all of the paths that have particle locations within a specified region in the path view for the frames currently displayed. The information displayed is the path number, the frame in which the path started, the length of the path and the start and end points.

Access

Access to the path list view is through popup menu 1 of the path view.

20.3 Popup menu

The Lagrangian field provides the following options in its popup menu.

- *Save as…* – saves the Lagrangian path field in a new file. All Lagrangian 2D path field files have the extension *.lgr2d*. The new file name is displayed next to the field name in the *Streams* object list and Lagrangian list views.
- *Save* – saves the Lagrangian 2D path field to disk. If the Lagrangian 2D path field currently has no associated file then the user is prompted for a file name. All Lagrangian 2D field files have the extension *.lgr2d*.
- *Save paths to text file…* – saves the paths to a text file specified by the user.
- *Create copy* – creates a copy of this Lagrangian path field and adds it to the *Streams* system. The copied path field includes all path line data but does not include the processes of the original path field. The copied path field has a default set of calculators containing no calculated fields.
- *Change name…* – allows the user to change the name of the Lagrangian path field. This name change will be reflected in the *Streams* object list and Lagrangian list views.
- *Edit description…* – allows the user to change the description of the velocity field time series.
- *Open calculator view* – opens the calculator view.
- *Open details view* – opens the details view.
- *Open path view* – opens the path view. This is the default view.
- *Open path list view* – opens the path list view.
- *Open process view* – opens the process view.
- *Execute process pipeline…* – executes the current process pipeline for the field. If this pipeline is invalid the execution will fail.
- *Close all views* – closes all open views for this object.
- *Close* – removes the Lagrangian field from *Streams*. All views are closed. If the Lagrangian field has unsaved data then the user is prompted to save this data before closure.

20.4 Calculators

The predefined calculators for a Lagrangian path field are listed in table 20.1. These calculators possess two independent variables, the path number and time.

Table 20.1 The list of available Lagrangian 2D path field calculators.

Symbol	Description
(u,v)	the velocity vector along a path
u	the x velocity component.
v	the y velocity component
(u',v')	the fluctuating velocity vector along a path where the average velocity is calculated by averaging over time along a path.
u'	the fluctuating u velocity where the average u velocity is calculated by averaging over time along a path.
v'	the fluctuating v velocity where the average v velocity is calculated by averaging over time along a path.
u'u'	the intensity of the fluctuating u velocity where the average u velocity is calculated by averaging over time along a path.
v'v'	the intensity of the fluctuating v velocity where the average v velocity is calculated by averaging over time along a path.
(x,y)	the particle locations long a path.
x	the x position of the particle along a path.
y	the y position of the particle along a path.
x rel	the change in the x position of the particle relative to its position at the start of the path.
y rel	the change in the y position of the particle relative to its position at the start of the path.
|r| rel	The distance travelled by the particle relative to its position at the start of the path.
(dx,dy)	the displacement vector between frames in a path. The first value is (0,0).
dx	the x displacement between frames. This value starts at 0 and is vectorial in that the direction is important in the calculation.
dy	the y displacement between frames. This value starts at 0 and is vectorial in that the direction is important in the calculation.

Symbol	Description
|dr|	the distance travelled between frames. This is always a positive number.
(dx',dy')	the perturbation displacement vector between frames. This is computed by subtracting the time average of (dx,dy) for each path from (dx,dy).
dx'	the x component of the perturbation displacement vector.
dy'	the y component of the perturbation displacement vector.
Two path	the two path calculator that enables quantities to be computed from two paths. See section 20.4.3.

20.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. The Lagrangian field calculator manager view allows the user to calculate any one of the 9 different projections that the field calculator provides. All field calculators have the same manager view, and therefore this view will be described in a generic way.

Each field calculator effectively generates a two-dimensional field from the Lagrangian path field object for which it is defined. The two dimensions are p, the path number, and t, time. The user is at liberty to extract a range of different projections of that two-dimensional field by specifying constraints on the two independent variables p and t. For each variable the user can select

- to leave that variable as a variable in the calculated field, specifying the range for that variable
- to average over that variable, specifying the range of that variable over which the average is to be calculated, or
- to choose a value for that variable, at which the field will be calculated.

The fields are given two names that indicate their genesis. The first, short name, matches the name of the calculator itself. The name of the second, longer name, begins with the field calculator name but this is followed by parentheses containing two groups of symbols, separated by

commas, referring to the variables, p and t respectively. These symbols are

- the variable name, if that variable was chosen to be variable, followed by its domain, or
- the variable name in angle brackets (<>) if that variable was averaged, followed by the limits for the averaging process, or
- the variable name, followed by its value, if that variable was fixed.

Finally the name is completed with the name of the Lagrangian field preceded by the word "from".

One or more fields can be selected and deleted by clicking the Delete field button, or they can be closed using the field object's popup menu.

When a Lagrangian field object is saved to disk all of its calculators and their calculated fields are saved with it. As these fields can be easily recalculated, if storage space is at a premium, the user may wish to delete all fields before saving the Lagrangian field object.

Access

The manager view is opened by double clicking on the field calculator in the Lagrangian field calculator view, or through the popup menu of the field calculator object.

20.4.2 Custom calculators

Lagrangian fields also provide the capability to define custom calculators. See section 20.2.1 for more details.

20.4.3 Two path calculators

A two path calculator provides significantly different tools to the other calculators listed in table 20.1. This calculator computes quantities based on the differences between two selected paths. The motivation for the provision of this calculator arises naturally from the domain of structural testing. Traditional measurement systems typically measure the differential displacement between two points on a structural specimen. For example, a potentiometer, designed to measure strain, measures the separation between the two points and converts this to a strain. The two path calculator enables the analyst to mimic these types of measurements by selecting two paths and treating them as the two ends of the potentiometer.

Seven different measurements, as functions of time, can be deduced from the two selected paths. These are: the difference in x and y positions, the distance between the two paths, the angle of the line connecting the two paths, the x and y strains, and the strain along the line connecting the two paths. A traditional potentiometer measures the third and last of these quantities.

20.4.4.1 Manager view

The Manager view for the two path calculator differs substantially from that described in section 20.4.1. The view includes three zones.

On the left the user can choose the two paths of interest by selecting their path numbers. If the path number isn't known then the Search button provides a simple mechanism to search for the path that lies closest to a particular point in space at a particular time. Pressing the Search button displays a dialog box where the user can specify the x, y and t values sought. Pressing the Find button in the dialog box yields the number of the closest path along with the actual x and y coordinates of the path at the specified time.

On the right seven buttons, corresponding to each of the measurements described above, are provided. Clicking on any of these will yield the quantity of interest for the two selected paths.

At the bottom of the view the calculated fields are displayed in a list box in the same way as for the manager view described in section 20.4.1. The user is able to delete fields which are no longer required.

20.4.4 Popup menu

All calculators provide the following popup menu

- *Open manager view* – opens the manager view. It is the default.

21 Lagrangian 3D Path Fields

21.1 Description

Lagrangian 3D path fields store particle path information derived from a particle 3D record. Particle matches must have been created for the record through the application of a PTV 3D analysis pipeline. A Lagrangian 3D path field contrasts with the Eulerian velocity 3D field described in chapter 13 in that flow information (actually particle information) is particle-based instead of being defined on a spatial grid. Particle location, velocity, radius and intensity are stored at each location along the particle path.

The data stored in a Lagrangian 3D path field can be thought of in two ways. Firstly as a set of particle paths, each of which may start at a different time, and end at a different time. Secondly as a two-dimensional matrix of data where the rows (or columns) are labeled by the path number and the columns (or rows) are labeled by time. This rectangular matrix will contain elements that are undefined because a particular path may not exist at that time. The paths are ordered by the time at which they start. Thus the earlier they begin the earlier they appear in the set of paths. However, there is no order imposed on paths that start at the same time.

Once created, a Lagrangian 3D path field object is displayed in the *Streams* object and Lagrangian field list views.

21.2 Views

The Lagrangian 3D path field provides seven views. These are

- calculator view
- details view
- particles view
- path view
- path list view
- process view
- selected path list view

21.2.1 Calculator view

Description

The calculator view is the default view. It lists all of the calculators available for the Lagrangian 3D path field. Currently 27 calculators are available and these are discussed in detail later in this chapter.

Access

Access to the calculator view is through the popup menu of the Lagrangian field object, or by double clicking the object.

Menus

Calculator Menu

- *Refresh calculators…* this option deletes all of the currently defined calculators (including their calculated fields) and replaces them with a new set of the predefined calculators. The user is prompted to confirm the action.
- *Create custom scalar calculator…* this option displays a dialog box that enables the user to create their own custom scalar field calculator (see below).
- *Create custom vector calculator…* this option displays a dialog box that enables the user to create their own custom vector field calculator (see below).
- *Delete selected calculators…* this option deletes all of the selected calculators from the intensity field object. The user is prompted to confirm the action.

21.2.1.1 Custom calculator dialog

When the user chooses to create a custom calculator a dialog box displayed. In contains the following tabs:

Name

On this tab the user can specify the name of the calculator. If the user leaves this field blank the formula text is substituted for the name. This name is displayed in the calculator list in the calculator view.

Formula

For a scalar field calculator a single text area is provided for the user to enter the formula for the calculator. This formula is defined by the rules

described in chapter 47. For a vector field calculator three text areas are provided for entering the formulae for the x, y and z components of the field.

21.2.2 Details view

Description

The details view gives information about the Lagrangian 3D path field. This information includes its name and description, details about the start and end time for the field, the number of paths, and the extent of the physical domain. Only the first two of these, along with the spatial and temporal units, can be changed within this view. The Save and Cancel buttons enable the user to save or discard changes made in this view.

Access

Access to the details view is through the popup menu of the Lagrangian field object.

21.2.3 Particles view

Description

The particles view displays information about a specific path within the Lagrangian field. All of the particle locations along the path are presented in tabular form. For each particle within the path the table lists its number, the time, its position, the two components of its velocity, its radius, intensity and area.

Access

Access to the particles view is through the popup menu of a path object. Paths are displayed in the path list view and the selected path list view of the Lagrangian field object.

21.2.4 Path view

Description

The path view has much in common with the particle 3D record particle view in that it displays particles and provides a simply video interface to observe the particle movement. However, the focus in this view is on paths. The user can select one or more frames to display starting at a particular time. All particles present in the selected frames are displayed.

The user may choose to display the paths associated with the particles present in the first frame displayed. The backward paths button displays the portion of the path before the first frame in yellow. The core paths button displays the path encapsulated by the frames currently displayed in red. And the forward paths button displays the portion of the path that exists beyond the last frame in blue. As the animation moves forward in time paths will come and go and the backward, core and forward portions will change. Path numbers can be displayed if desired.

Particles on each path can be displayed as points, or in their true size, shape and orientation. In addition the particles can be drawn in their true colour or in monochrome, and the particle velocities can be indicated with vectors. Thus most of the particle data are available to the user.

The view has five toolbars displayed above the path display area. These toolbars give the user considerable control over the operation and appearance of the view. These toolbars are described below.

Animation toolbar

The top toolbar is the animation toolbar. This toolbar provides the video controls that allow the user to control the animation features of the view.

- *Current time* – this combo box displays the first frame currently displayed as well as its time. The user can select any value in this box to move to a different time in the field.
- *Number of frames displayed* – this combo box selects the number of frames (and corresponding time) overlaid in the view.
- *Go to next time* – steps the display forward one frame.
- *Go to previous time*– steps the display back one frame.
- *Play*– starts the animation forward in time.
- *Play in reverse* – starts the animation backward in time.
- *Pause* – pauses the animation.
- *Go to start* – resets the display to the first frame.
- *Go to end* – resets the display to the last frame.
- *Animation timing delay (ms)* – this text box allows the user to change the speed of the animation. The displayed timing delay is the time step for the record. The actual delay may be longer if the drawing overhead is high.

Path toolbar

The path tool bar sits below the animation toolbar. It allows the user to select the paths, particles and vectors that are displayed in the view.

- *Particles* – when this button is pressed all the particles in the visible paths are displayed.
- *Forward paths* – displays the forward paths
- *Core paths* – displays the core paths
- *Backward paths* – displays the backward paths
- *Vectors* – displays the velocity vectors at each point on each path.
- *Numbers* – displays the path numbers at the start of each path.

Cursor toolbar

The cursor toolbar, second from the bottom, displays the location of the cursor on screen and indicates the current drawing tool.

- *x* – This text box displays the x coordinate (in the specified spatial units) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *y* – This text box displays the y coordinate (in the specified spatial units) of the current cursor position. The box is updated as the cursor is moved over the video frame area. It cannot be edited by the user.
- *Drawing tool* – This text box displays the name of the currently selected drawing tool. This tool is used for defining regions in the view. The tool can be changed through the *Region* menu.

x toolbar

- *x min* – This text box displays the minimum value of the x axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- x max – This text box displays the maximum value of the x axis currently displayed.
- *Focal x* – This text box displays the x coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

y toolbar

- *y min* – This text box displays the minimum value of the y axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- y max – This text box displays the maximum value of the y axis currently displayed.

- *Focal y* – This text box displays the y coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

z toolbar

- *z min* – This text box displays the minimum value of the z axis currently displayed. This can be changed by the user, thus allowing them to restrict the portion of each frame that is visible. It should be noted that the frame is always drawn with the correct aspect ratio – thus units in the x, y and z directions have the same length on screen.
- z max – This text box displays the maximum value of the z axis currently displayed.
- *Focal z* – This text box displays the z coordinate of the current cursor position in the focal plane. The box is updated as the cursor is moved over the display area. It cannot be edited by the user.

The main menu bar provides the following menus.

Menus

File Menu

- *Save as image*… – allows the user to save the path view (as it is currently displayed and without the toolbars or defined regions) as an image using one of the standard image formats – TIFF, JPEG, PNG and BMP. The colours currently selected for the view are used in the image. The user has considerable control over the stored image including choice of font size, line thickness, resolution etc, and is able to preview the image before it is stored on disk. Further details of this process are provided in the User's Guide: Field Objects.
- *Save as image sequence*… allows the user to save each of the frames in the path field as an image in an image sequence. These images will be stored in one of the standard image formats – JPEG, PNG and BMP – as selected by the user. The user can select the resolution of the images to be used. The user specifies a name for the image files and *Streams* automatically adds a 5 digit number (starting at 0) to each file name. Further details of this process are provided in the User's Guide: Field Objects.

View Menu

- *Set layout*… – – this displays a dialog box that enables the user to set a range of display parameters common to all Lagrangian 3D path field path views (i.e. changing these settings for this view changes them for all path views). Details regarding this dialog box are provided in section 21.2.4.1).
- *Draw point particles*… – a check menu item that determines whether the particles are displayed as points.

- *Draw true particles…* – a check menu item that determines whether the particles are displayed as their true shapes and sizes.
- *Draw monochrome particles…* – a check menu item that determines whether the particles are drawn in white.
- *Draw true colour particles…* – a check menu item that determines whether the particles are drawn using their true colour.
- *Show particles…* – a check menu item that determines whether the particles are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show forward paths* – a check menu item that determines whether the forward paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show core paths* – a check menu item that determines whether the core paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show backward paths* – a check menu item that determines whether the backward paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show vectors* – a check menu item that determines whether the vectors paths are displayed. This has the same effect as the equivalent button in the paths toolbar.
- *Show path numbers* – displays the path numbers at the start of each path.
- *Show frame* – displays the frame around the particle domain.
- *Show scales*– displays the scales on the particle domain.
- *Show regions*– displays all defined regions.
- *Redraw during resize* – if set, the view redraws during a resize operation. If the drawing operation is extensive, for example many frames with large numbers of particles, this might cause a lack of responsiveness. If not set redrawing does not occur and the user needs to perform some manual operation after the resize in order to force a redraw.
- Change variable decimal places… – this item opens a dialog box in which the number of decimal places used to display the x, y, z and t variables can be set. Sensible minimum values are set for these.
- Reset min/max to default – this resets the x, y and z axis ranges to their defaults.
- Hide toolbars – when selected the toolbars at the top of the view are hidden. This provides increased display space. The majority of the controls in the toolbars are available through the popup menu or one of the menus in the menubar. This can also be effected by double clicking the mouse while holding down the ALT key.

Animation Menu

- *Go to next time…* – steps the display forward one frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to previous time…*– steps the display back one frame. This has the same effect as the equivalent button in the animation toolbar.

- *Play*…– starts the animation forward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Play in reverse* – starts the animation backward in time. This has the same effect as the equivalent button in the animation toolbar.
- *Pause* – pauses the animation. This has the same effect as the equivalent button in the animation toolbar.
- *Go to start* – resets the display to the first frame. This has the same effect as the equivalent button in the animation toolbar.
- *Go to end* – resets the display to the last frame. The user is prompted to confirm this option. This has the same effect as the equivalent button in the animation toolbar.

Cameras menu

- *Manage cameras*… – opens the camera management view. See chapter 2 of the *User's Guide: Field Objects* manual.
- *Open current camera controller* – this opens the controller view of the currently selected camera so the user can make adjustments to the camera settings. These changes are often most easily achieved using keyboard commands (see chapter 2 of the *User's Guide: Field Objects* manual). The camera controller view is also opened by double clicking the mouse on the display area with SHIFT and ALT keys held down.

Lighting menu

- *Manage lighting*… – this option opens the manage lighting view. See chapter 3 of the *User's Guide: Field Objects* manual.

Region Menu

- *Manage regions*… – displays region 3D manager view (see chapter 48) and enables the user to create 3D regions in the view.
- *Load regions*…– previously stored regions are loaded from a disk file and added to the currently defined regions. Regions are stored in a file with a *.rgn3* extension.
- *Save regions as*…– saves all of the currently defined regions in a *.rgn3* disk file.
- *Delete all regions* – deletes all currently defined regions. The user is prompted to confirm this option.
- *Delete selected regions* – deletes all selected regions.
- *Select all regions* – selects all regions.
- *Select <-> unselect* – makes all selected regions unselected and vice versa.
- *Invert selected regions* – converts all selected regions that are interior regions into exterior regions and vice versa.
- *Invert all regions* – converts all regions that are interior regions into exterior regions and vice versa.

- *Delete particles in selected regions* – this will delete all particles that lie within the selected 3D regions for the frames currently displayed. The user will be asked to confirm the action.

Zoom Menu

- *Zoom* – causes the view to be zoomed using the currently displayed zoom rectangle.
- *Unzoom* – this undoes any zoom operation by redisplaying the entire video frame area.
- *Clear zoom rectangle*– this undoes any zoom operation by redisplaying the entire video frame area.

Two popup menus are provided with this view. The first of these is discussed under the section on regions (21.2.4.2). When the user right clicks the mouse button within a rectangle drawn on the screen this menu is displayed. However a second popup menu is displayed if the user right clicks anywhere on the display that is not within this rectangle. This popup menu contains items that replicate those in the main menu bar and is provided simply as an alternative means of accessing this functionality.

Popup Menu

- *Set layout…* – see View menu.
- *Go to next time* – see Animation menu.
- *Go to previous time* – see Animation menu.
- *Play* – see Animation menu.
- *Play in reverse* – see Animation menu.
- *Pause* – see Animation menu.
- *Go to start* – see Animation menu.
- *Go to end* – see Animation menu.
- *Show particles…* – see View menu.
- *Show forward paths* – see View menu.
- *Show core paths* –see View menu.
- *Show backward paths* – see View menu.
- *Show vectors* – see View menu.
- *Show path numbers* – see View menu.
- Show frame – see View menu.
- Show scales – see View menu.
- Show regions – see View menu.

21.2.4.1 Set layout dialog

The user has considerable control over the appearance of this view. The set layout dialog can be opened through the View menu or a popup menu. It is important to note that the parameters set in this dialog box apply to all particle views so that when other views are opened they will possess the same settings. The three buttons at the bottom of the dialog are:

- Reset… – resets all of the layout settings to their defaults.
- *OK* – confirms the changes made and applies them to the view.
- *Cancel*…– cancels the changes and closes the dialog box.

The dialog includes 3 tabs – General, Borders and Colour.

General

- *Number of x divisions* – this parameter specifies how many divisions appear along the x axis and hence how many scale values will be displayed.
- *Number of y divisions* – this parameter specifies how many divisions appear along the y axis and hence how many scale values will be displayed.
- *Number of z divisions* – this parameter specifies how many divisions appear along the z axis and hence how many scale values will be displayed.
- *Radius factor* – when the particles are drawn as spheres their radii are multiplied by this factor. This gives the user some control over how large the particles appear in the view.
- *Geometric scale factor* – this option allows the user to globally change the "size" of objects such as the axes thickness.
- *Font size* – sets the font size. The default is 14.
- *Draw axes at origin* – the default setting is to draw the axes to the left, bottom and back of the domain. By selecting this checkbox the axes are relocated so they pass through the origin.

Borders

- *Left border width* – the view includes a viewable region surrounded by borders in which the axis labels and scales appear as well the info label in the centre at the bottom. This setting is for the size, in pixels, of the left hand border. It is worth noting that the size of the borders may need to change when producing an image of this view depending on the resolution chosen for the image.
- *Right border width* – sets the size, in pixels, of the right hand border.
- *Bottom Border width*– sets the size of the bottom border in pixels.
- *Top border width* – sets the size of the top border in pixels.

Colour

- *Points colour* – the colour of point particles. It can be changed by clicking on the Select… button.
- *Axes colour* – the colour of axes. It can be changed by clicking on the Select… button.
- *Labels colour* – the colour of all labels, including the scales, axis names and the info text at the bottom of the view. It can be changed by clicking on the Select… button.
- *Forward paths colour* – the colour of lines for forward paths. It can be changed by clicking on the Select… button.
- *Core paths colour* – the colour of lines for core paths. It can be changed by clicking on the Select… button.
- *Backward paths colour* – the colour of lines for backward paths. It can be changed by clicking on the Select… button.
- *Velocity vector colour* – the colour used for drawing velocity vectors. It can be changed by clicking on the Select… button.
- *Background colour* – the background colour. It can be changed by clicking on the Select… button.

21.2.4.2 Regions

Both two-dimensional (see chapter 47) and three-dimensional regions (see chapter 48) are used in a number of ways in this view. Three-dimensional regions, which are created and managed through the Region menu are typically used for removing particles from the record. Unlike 2D regions these are not easily drawn on screen and thus must be defined by entering their parameters. Two-dimensional rectangular regions are used for zooming in and out in the view, and for selecting particles whose details are of interest. In both cases these features are enabled by drawing rectangular regions on the display in the usual way (see chapter 47). The two-dimensional rectangle is always interpreted as residing in the focal plane of the current camera.

When the user right clicks the mouse in a 2D rectangular region the following popup menu is displayed.

- *Deletes paths in region…* – deletes the particle paths that cross the region in the frames currently displayed.
- *Open selected particle list view…* – displays the selected particle list view (see section 21.2.7) based on the particles and frames displayed within the drawn rectangle.
- *Zoom…* – zooms the view.
- *Unzoom…* – this undoes any zoom operation.

Access

Access to the path view is through the popup menu of the Lagrangian 3D path field.

21.2.5 Path list view

Description

The path list view lists all of the paths in the field in tabular form. The table includes the following data associated with a path: its number, the frame in which it starts, its length (in frames), the start and end points, the physical length of the path or the distance travelled by the particle, the total displacement from the start point to the end point (a vector), the mean velocity of the particle along the path, the mean radius of the particle along its path, and the standard deviations of the velocity and radius along its path.

Details of particular paths can be obtained by righting clicking on a path in the list and choosing to open the particles view.

Menus

File Menu

- *Save to text file…* – saves the paths (a subset of which can be selected by the user) to a text file with a .txt or .csv extension. The text file provides the following information for each path line – number, start frame, length, x and y coordinates of the start and end point of the path, total distance travelled, net displacement in the x and y directions, the average u and v velocities, the average radius of the particle along the path, and the standard deviations of u, v, and radius along the path.

View Menu

- *Change decimal places…* – changes the number of decimal places used to display the parameters in the table.

Sort Menu

- *Sort by number* – sorts the path table by the path number.
- *Sort by start x* – sorts the path table by the initial x position.
- *Sort by start y* – sorts the path table by the initial y position.
- *Sort by start z* – sorts the path table by the initial z position.
- *Sort by end x* – sorts the path table by the last x position.

- *Sort by end y* – sorts the path table by the last y position.
- *Sort by end z* – sorts the path table by the last z position.
- *Sort by x disp* – sorts the path table by the x displacement along the path.
- *Sort by y disp* – sorts the path table by the y displacement along the path.
- *Sort by z disp* – sorts the path table by the z displacement along the path.

Access

Access to the path list view is through the popup menu of the Lagrangian 3D path field object.

21.2.6 Process view

Description

The process view enables the user to interact with the processes defined for the field object, and to create and load new ones. This view is the same for all entities and details are provided in section 3.2.3.

Access

Access to the calculator view is through the popup menu of the Lagrangian path field object.

21.2.7 Selected path list view

Description

The selected path list view lists all of the paths that have particle locations within a specified region in the path view for the frames currently displayed. The information displayed is the path number, the frame in which the path started, the length of the path and the start and end points.

Access

Access to the path list view is through the first popup menu of the path view.

21.3 Popup menu

The Lagrangian 3D path field provides the following options in its popup menu.

- *Save as…* – saves the Lagrangian 3D path field in a new file. All Lagrangian 3D path field files have the extension *.lgr3d*. The new file name is displayed next to the field name in the *Streams* object list and Lagrangian list views.
- *Save* – saves the Lagrangian 3D field to disk. If the Lagrangian field currently has no associated file then the user is prompted for a file name. All Lagrangian 3D field files have the extension *.lgr3d*.
- *Save paths to text file…* – saves the paths to a text file specified by the user.
- *Create copy* – creates a copy of this Lagrangian path field and adds it to the *Streams* system. The copied path field includes all path line data but does not include the processes of the original path field. The copied path field has a default set of calculators containing no calculated fields.
- *Change name…* – allows the user to change the name of the Lagrangian path field. This name change will be reflected in the *Streams* object list and Lagrangian list views.
- *Edit description…* – allows the user to change the description of the velocity field time series.
- *Open calculator view* – opens the calculator view.
- *Open details view* – opens the details view.
- *Open path view* – opens the path view. This is the default view.
- *Open path list view* – opens the path list view.
- *Open process view* – opens the process view.
- *Execute process pipeline…* – executes the current process pipeline for the field. If this pipeline is invalid the execution will fail.
- *Close all views* – closes all open views for this object.
- *Close* – removes the Lagrangian 3D path field from *Streams*. All views are closed. If the Lagrangian field has unsaved data then the user is prompted to save this data before closure.

21.4 Calculators

The predefined calculators for a Lagrangian 3D path field are listed in table 21.1. These calculators possess two independent variables, the path number and time.

Table 21.1 The list of available Lagrangian 3D path field calculators.

Symbol	Description
(u,v,w)	the velocity vector along a path
u	the x velocity component.
v	the y velocity component
w	the z velocity component

Symbol	Description
(u',v',w')	the fluctuating velocity vector along a path where the average velocity is calculated by averaging over time along a path.
u'	the fluctuating u velocity where the average u velocity is calculated by averaging over time along a path.
v'	the fluctuating v velocity where the average v velocity is calculated by averaging over time along a path.
w'	the fluctuating w velocity where the average w velocity is calculated by averaging over time along a path.
u'u'	the intensity of the fluctuating u velocity where the average u velocity is calculated by averaging over time along a path.
v'v'	the intensity of the fluctuating v velocity where the average v velocity is calculated by averaging over time along a path.
w'w'	the intensity of the fluctuating w velocity where the average w velocity is calculated by averaging over time along a path.
(x,y,z)	the particle locations long a path.
x	the x position of the particle along a path.
y	the y position of the particle along a path.
z	the z position of the particle along a path.
x rel	the change in the x position of the particle relative to its position at the start of the path.
y rel	the change in the y position of the particle relative to its position at the start of the path.
z rel	the change in the z position of the particle relative to its position at the start of the path.
|r| rel	The distance travelled by the particle relative to its position at the start of the path.
(dx,dy,dz)	the displacement vector between frames in a path. The first value is (0,0).
dx	the x displacement between frames. This value starts at 0 and is vectorial in that the direction is

Symbol	Description
	important in the calculation.
dy	the y displacement between frames. This value starts at 0 and is vectorial in that the direction is important in the calculation.
dz	the z displacement between frames. This value starts at 0 and is vectorial in that the direction is important in the calculation.
|dr|	the distance travelled between frames. This is always a positive number.
(dx',dy',dz')	the perturbation displacement vector between frames. This is computed by subtracting the time average of (dx,dy,dz) for each path from (dx,dy,dz).
dx'	the x component of the perturbation displacement vector.
dy'	the y component of the perturbation displacement vector.
dz'	the z component of the perturbation displacement vector.
Two path	the two path calculator that enables quantities to be computed from two paths. See section 21.4.3.

21.4.1 Manager View

Description

Each calculator provides a single view known as the manager view. The Lagrangian 3D path field calculator manager view allows the user to calculate any one of the 9 different projections that the field calculator provides. All field calculators have the same manager view, and therefore this view will be described in a generic way.

Each field calculator effectively generates a two-dimensional field from the Lagrangian 3D path field object for which it is defined. The two dimensions are p, the path number, and t, time. The user is at liberty to extract a range of different projections of that two-dimensional field by specifying constraints on the two independent variables p and t. For each variable the user can select

- to leave that variable as a variable in the calculated field, specifying the range for that variable
- to average over that variable, specifying the range of that variable over which the average is to be calculated, or
- to choose a value for that variable, at which the field will be calculated.

The fields are given two names that indicate their genesis. The first, short name, matches the name of the calculator itself. The name of the second, longer name, begins with the field calculator name but this is followed by parentheses containing two groups of symbols, separated by commas, referring to the variables, p and t respectively. These symbols are

- the variable name, if that variable was chosen to be variable, followed by its domain, or
- the variable name in angle brackets (<>) if that variable was averaged, followed by the limits for the averaging process, or
- the variable name, followed by its value, if that variable was fixed.

Finally the name is completed with the name of the Lagrangian 3D path field preceded by the word "from".

One or more fields can be selected and deleted by clicking the Delete field button, or they can be closed using the field object's popup menu.

When a Lagrangian 3D path field object is saved to disk all of its calculators and their calculated fields are saved with it. As these fields can be easily recalculated, if storage space is at a premium, the user may wish to delete all fields before saving the Lagrangian field object.

Access

The manager view is opened by double clicking on the field calculator in the Lagrangian 3D path field calculator view, or through the popup menu of the field calculator object.

21.4.2 Custom calculators

Lagrangian 3D path fields also provide the capability to define custom calculators. See section 21.2.1 for more details.

21.4.3 Two path calculators

A two path calculator provides significantly different tools to the other calculators listed in table 21.1. This calculator computes quantities based on the differences between two selected paths. The motivation for the provision of this calculator arises naturally from the domain of structural testing. Traditional measurement systems typically measure the differential displacement between two points on a structural specimen. For example, a potentiometer, designed to measure strain, measures the separation between the two points and converts this to a strain. The two path calculator enables the analyst to mimic these types of measurements by selecting two paths and treating them as the two ends of the potentiometer.

Eight different measurements, as functions of time, can be deduced from the two selected paths. These are: the difference in x, y and z positions, the distance between the two paths, the x, y and z strains, and the strain along the line connecting the two paths. A traditional potentiometer measures the third and last of these quantities.

21.4.4.1 Manager view

The Manager view for the two path calculator differs substantially from that described in section 21.4.1. The view includes three zones.

On the left the user can choose the two paths of interest by selecting their path numbers. If the path number isn't known then the Search button provides a simple mechanism to search for the path that lies closest to a particular point in space at a particular time. Pressing the Search button displays a dialog box where the user can specify the x, y, z and t values sought. Pressing the Find button in the dialog box yields the number of the closest path along with the actual x, y and z coordinates of the path at the specified time.

On the right eight buttons, corresponding to each of the measurements described above, are provided. Clicking on any of these will yield the quantity of interest for the two selected paths.

At the bottom of the view the calculated fields are displayed in a list box in the same way as for the manager view described in section 21.4.1. The user is able to delete fields which are no longer required.

21.4.3 Popup menu

All calculators provide the following popup menu

- *Open manager view* – opens the manager view. It is the default.

22 Free Processes

22.1 Description

Free processes exist outside of an entity and are accessible directly from the application's main menu. The free processes available in *Streams* fall into four categories:

- processes associated with image sequence creation
- processes that combine particle records or use theoretical models to create artificial particle records
- processes that combine Eulerian fields, or use theoretical models to create them and
- a batch process.

These will be discussed in the following sections.

Free processes can be saved to disk and loaded from disk via the process menu of the main application view. Once loaded these processes are listed in the object and process list views and can be manipulated through their popup menus.

22.1.1 Popup menu

All free processes provide the following popup menu

- *Save as…* – saves the process in a new file with an extension of .pcf.
- *Save* – saves the process.
- *Change name…* – allows the user to change the process name.
- *Open details view* – opens the details view for the process.
- *Execute* – executes the process.
- *Close all views* – closes all open views.
- *Close* – closes the process and removes it from the system.

22.2 Image processes

Currently there are four process is associated with images and image sequences.

22.2.1 Overlay images process

In sophisticated optical experimental systems it is not unusual to need to stitch images together in some form. This process provides this capability. The process is defined by a set of overlays. Each overlay includes one or more images (all of the same size) together with an x and y offset (measured in pixels) and possibly a list of two-dimensional regions. Each overlay must contain the same number of images. The user must also choose the type of data to be processed in the images and specify an output file name.

The process executes as follows. One image is taken from each overlay in the order they are defined. Each image is offset from a global origin by the offsets defined for the overlay. The images are then overlaid on each other. If the images overlay one another then the pixel value at any point is an average of the pixels from the contributing images. If an overlay has any regions defined only pixels that lie within those regions are included in the overlay process. Once the stitched imaged is complete it is saved to the output file where the file name has a five digit number added to. The next image in each overlay is then processed in the same way until all images in each overlay have been processed.

To summarise, if the process includes *n* overlays and each overlay contains *m* images, then the result will be *m* stitched images, each containing some or all of the contents of *n* images.

When the overlay image process is created the user is presented with a dialog box that provides the following options. The details view of the process includes the same options.

The dialog box has three tabs, Process, Details and Overlays

Process

- *Process name* – the name of the auto generation process.

Details

- *Output file* – the name of the output stitched image file. This name will have a five digit number added to the end of the file name to differentiate between the different overlays produced by the process.
- *Browse* – this button allows the user to select the output directory, file name and file type.
- *Image type* – the type of data contained in the images.

Overlays

- *Overlays* – this list box lists all of the overlays defined for this process. Each overlay has a name defined for it and it it this name that is displayed in the list box. a five digit number added to the end of the file name to differentiate between the different overlays produced by the process.
- *Add..* – this button allows the user to add a new overlay. A dialog box is displayed that enables the user to specify the following data: Name – the name of the overlay; x offset (pix) – the x offset in pixels. The image is shifted to the right by this number; y offset (pix) – the y offset in pixe;s. The image is shifted up this amount; Images – the user can add and delete images using the load and delete buttons. All images are displayed in the list box; Regions – the list of regions that define the pixels to be included in the overlay. If no regions are defined all pixels are included. The load and delete buttons allow the user to populate and edit the list of regions.
- *Delete* – this button allows the user to delete overlays from the list.

22.2.2 Average image sequences process

Averaging sequences of images is a common requirement of experimental optical systems. For example in a LIF system calibration images are often obtained by averaging the image of a calibration cell over a period of time to reduce random noise.

The average image sequence process automates the averaging of multiple sets of images. It accomplishes the same task as the average image sequence process that can be defined for an image sequence. However that process requires the construction of each image sequence to be averaged and for each averaging process to be defined and executed separately. The end result is often the discarding of the image sequences only the average images were of interest in the first place.

For this process the user places each set of images to be averaged into a separate directory (the directory should include no other files) and then selects those directories into the process. An output directory is defined as well as the image data type of images and the file format to be used for the average images.

The process executes as follows. Each directory is processed in turn. The images in each directory are loaded and averaged and the resulting average image is output into the output directory with a name that corresponds to its source directory name and with a file format as specified by the user. The process executes until all directories have been

processed. The number of average image files produced matches the number of directories specified by the user.

This process can be created and then left to run overnight if large numbers of images are involved.

When the average image sequences process is created the user is presented with a dialog box that provides the following options. The details view of the process includes the same options.

Process

- *Process name* – the name of the average image sequences process.

Details

- *Output directory* – the directory in which all of the averaged images iare to be placed. It is important to remember that it is the directory that needs to be selected here not a file within the directory.
- *Browse* – this button allows the user to navigate the directory structure and select the output directory.
- *Input image type* – the type of data contained in the images.
- *Output file format* – the file format of the averaged image files.

Directories

- *Directory list* – the list of directories containing the images to be averaged. Each directory must include only the image files for processing.
- *Add…* – this button allows the user to add a directory to the list.
- *Delete* – this button allows the user to delete a directory from the list.

22.2.3 Image sequence auto generation process

Image sequences are normally the starting point for analysis using *Streams*. It is not unusual, once an experimental system has been finally implemented and calibrated, that many experimental runs can be completed with the same experimental configuration. From a *Streams* perspective each run is likely to have the same processing requirements and the image sequence that corresponds to each run will have almost identical parameters and processes.

The image sequence auto generation process is designed to automate the creation of virtually identical image sequences that may then be processed en masse using the batch process described below.

It is assumed that the images captured in this set of runs are stored in a particular way on disk. The user will create a parent directory in the file system and within this parent directory a sub-directory will be created for each experimental run, and within each of these sub-directories will reside the images that belong to the corresponding image sequence. These images are assumed to all be of the same size, although they don't necessarily need to be of the same format (e.g. BMP, PNG etc.). The directory may contain other data files (which can't be image files) and these are ignored by the process.

The process will create an image sequence for each sub-directory within the parent, adding all of the image files that reside within the directory to the sequence. The process will have standard image sequence parameters, such as x and y scales, time step and the like defined for it and these will be used as the initialization parameters for all of the image sequences created. In addition the user can specify a set of bound processes, with a corresponding pipeline, that will be defined for all image sequences created. Once the process completes, a number of new image sequences will have been created and stored in the parent directory. Each image sequence will possess the same parameters, the same processes (see below), but different images.

The processes for each image sequence must be carefully defined. Many processes have some type of output that has both a name, and perhaps a file in which it is to be stored. For example a particle identification process will create a particle record that has a user-defined name. A save velocity field process will have a user defined filename for the file in which the field will be stored. Similarly an image sequence transform, or a filter pipeline, if the last process in a pipeline, will create processed images and store them to disk. The image sequence auto generation process takes particular care with these names. It will substitute the name of the image sequence for the name specified in the generic process defined for the image sequence auto generation process when the image sequence is created. So, for example, if a new image sequence has the name Run_1_2 and one of the processes defined for the sequence is a particle identification process, the particle record name defined for this process will be Run_1_2. Similarly if the sequence has a save velocity 2D field process the name of the file in which the velocity field will be stored will be Run_1_2.vel2d.

In order to manage the files that will be created in processing the set of newly created image sequences the auto generation process creates a directory called *output* in each of the sub-directories within the parent. It is in these directories that the output files, whether entity files such as a particle record, or image files generated by a filter pipeline or image sequence transform, are stored.

When the auto generation process is created the user is presented with a dialog box that provides the following options. The details view of the process includes the same options.

Process

- *Process name* – the name of the auto generation process.

Sequence

- *Image type* – the type of data contained in the images.
- *Temporal unit* – a text string designating the temporal unit.
- *Time step* (unit) – the time step for all image sequences created.
- *Pulsed time step (unit)* – the pulsed time step if all sequences are pulsed.
- *Spatial unit* – a text string designating the spatial unit.
- *x scale (unit/pix)* – the conversion from pixels to mm in the horizontal.
- *y scale (unit/pix)* – the conversion from pixels to mm in the vertical.
- *Pulsed* – check to specify a pulsed sequence of images.
- *Use directory name* – if checked the name of the sub-directory will be used for the image sequence associated with it.
- *Use following name + number* – if checked the name given in the following text box will be used to name the image sequences. These names will have a number added to them in order to ensure uniqueness.
- *Parent directory* – the parent directory in which all image sub-directories reside.

Processes

The top list box contains the processes that will be created for all image sequences. There are six buttons beneath the process list. These are

- *New* – displays a dialog box from which the user can select a new process. The dialog box presents two lists, the one on the left contains the primary processes for the entity and the right hand box the secondary processes. When a new process is selected from the box an initialization dialog box for that entity is displayed.
- *Load* – loads a previously defined bound process from disk. Bound processes have the file extension .pcb.

- *Delete* – deletes the currently selected process. The user is asked to confirm the action.
- *Clear* – deletes all currently defined processes. The user is prompted to confirm this action.
- *Add to pipeline* – adds the currently selected process to the end of the process pipeline. If a process is already a component in the pipeline it cannot be added again.
- *Insert in pipeline* – inserts the selected process before the currently selected (highlighted) process in the pipeline. Again, a process cannot be added more than once.

Process pipeline

The bottom list box contains the currently defined pipeline processes that will be created for all image sequences. There are two buttons beneath the pipeline process list. These are

- *Delete* – deletes the currently selected process from the pipeline. If this process is present in the process list it is not deleted from that list. The user is asked to confirm the action.
- *Clear* – deletes all currently defined processes from the pipeline. This action does not remove them from the list of processes. The user is prompted to confirm this action.

22.2.4 Create constant colour image process

This simple process produces an image with a constant intensity for all pixels. The image can be of any data type and the user can choose the image size and the intensity value to be used.

Process

- *Process name* – the name of the create constant colour image process.

Details

- *Image type* – the type of data contained in the images.
- *Width (pixels)* – the image width in pixels.
- *Height (pixels)* – the image height in pixels.
- *Red intensity (0-255)* – the intensity of the red channel. Used if the image is of type RGB.
- *Green intensity (0-255)* – the intensity of the green channel. Used if the image is of type RGB.
- *Blue intensity (0-255)* – the intensity of the blue channel. Used if the image is of type RGB.
- *Gray scale 10 bit intensity (0-1023)* – the intensity of the blue channel. Used if the image is of type gray scale 10 bit.
- *Gray scale 12 bit intensity (0-4095)* – the intensity of the blue channel. Used if the image is of type gray scale 12 bit.

- *Real value* – the real value of each pixel. Used if the image is of type real.
- *File name* – the full path for the file in which to store the image.
- *Browse* – pressing this button allows the user to navigate the directory structure and specify the file for saving the image. The user selects the file format (JPG etc) during this process.

22.3 Particle record processes - models

Model particle records are created by selecting particle record processes from the process menu, and choosing the appropriate model process from the submenu (the available models are discussed below and in the *System Theory and Design* Manual). Models can be used to construct particle 2D records and particle 3D records. In the latter case there is a process that creates a full three dimensional particle field and another process that creates a surface of particles in 3D space - the second of these is designed to produce displacement 3D surface fields.

The user is presented with a dialog box for the process creation. Five tabs, Process, Record, Particle, Matching and Integration, prompt the user for generic information about the process and its underlying model, while one tab, Flow, provides flow/model specific information. The data to be provided in each of these tabs is discussed below. Note that some parameters only arise in certain processes and these are clearly marked.

Process

On this tab the user provides the name of the process.

- *Create model process name* – the name of the process. This is given a default name that includes the name of the theoretical model.

Record

The user makes choices regarding the record that will be created by the process in this tab. This information is associated with the record itself, and is similar to that specified when a record is generated from a sequence of images.

The following parameters are set here:

- *Record name* – the name given to the particle record.
- *Width (mm)* – the width of a frame in mm.

- *x min (mm)* – (only used for the creation of particle 3D records where all particles lie on a surface) the minimum x value used for particle coordinates in mm.
- *x max (mm)* – (only used for the creation of particle 3D records where all particles lie on a surface) the maximum x value used for particle coordinates in mm.
- *Height (mm)* – the height of a frame in mm.
- *y min (mm)* – (only used for the creation of particle 3D records where all particles lie on a surface) the minimum y value used for particle coordinates in mm.
- *y max (mm)* – (only used for the creation of particle 3D records where all particles lie on a surface) the maximum y value used for particle coordinates in mm.
- *Depth (mm)* – (only provided for the creation of particle 3D records) the depth of a frame in mm.
- *Time step* (s) – the time step between frames in seconds, or between every second frame if the record is pulsed
- *Number of frames* – the total number of frames to generate for this record.
- *Clip is pulsed* – (not available for the creation of particle 3D records) select if the record is comprised of pairs of frames separated by a different time step
- *Pulse time step (s)* – (not available for the creation of particle 3D records) the time step between frames within a pulsed pair.

Particle

This tab prompts the user for information about the particles that are to be created in each frame. Note that not all particle properties are created in these artificial records. For example particle intensity matrices and particle shape information are not available.

The following parameters are set here:

- *Number of particles* – number of particles per frame
- *Loss/gain fraction* – the fraction of the particles that randomly leave the flow in each frame (0.0 means all particles remain in the flow (unless they have actually travelled through a frame boundary), and 1.0 means every particle disappears in every frame
- *Minimum radius (mm)* – particles are given a radius that is randomly distributed between two limits and they retain this value in all frames. This is the lower limit.
- *Maximum radius (mm)* – particles are given a radius that is randomly distributed between two limits and they retain this value in all frames. This is the upper limit.
- *Minimum intensity (0-255)* – particles are given an intensity that is randomly distributed between two limits, and they retain this value in all

frames. This is the lower limit. If the user chooses to have colour particles each colour intensity uses this value.

- *Maximum intensity (0-255)* – particles are given an intensity that is randomly distributed between two limits, and they retain this value in all frames. This is the upper limit. If the user chooses to have colour particles each colour intensity uses this value.
- *Use colour*– if this is selected, particles with different red, green and blue intensities are created. If not selected all particle are monochrome.
- *Reseed particles* – (not available for processes that create particle 3D records with particles on a surface) if this is selected, particles that leave the flow are reintroduced so that the total number of particles in each frame remains constant.
- *Distribute uniformly* – (only available for processes that create particle 2D records) if this is selected, particles are uniformly distributed throughout the frame. If not, then the particles are more densely seeded near the origin, with decreasing density as *x*, and *y* increase.

Flow

Each model flow provides a different tab for setting the flow parameters. The eight models and their parameters to be set in this tab are described below. For more complete information of the theoretical models please refer to the *System Theory and Design* Manual.

Cellular model

This model creates a laminar cellular flow specified by a streamfunction of the following form

$$\psi = A \sin\left(\frac{n\pi x}{w}\right) \sin\left(\frac{m\pi y}{h}\right) \qquad (22.1)$$

Here w and h are the width and height of the flow in mm. A, n and m are constants set by the user – note that m and n are integer values, and A is in mm^2/s.

Irregular cellular model

This second model is a somewhat more complex version of the cellular model. Two different cells are superimposed on a uniform flow in the x direction. In addition the cells have an oscillatory time dependence.

$$\psi = A_1 \sin\left(\frac{n_1 \pi x}{w}\right)\sin\left(\frac{m_1 \pi y}{h}\right)\cos\left(\omega_1 t\right) + A_2 \sin\left(\frac{n_2 \pi x}{w}\right)\sin\left(\frac{m_2 \pi y}{h}\right)\cos\left(\omega_2 t\right) + Uy \qquad (22.2)$$

Here w and h are the width and height of the flow in mm. A_1, n_1, m_1,ω_1, A_2, n_2, m_2,ω_2 and U are constants set by the user – note that n_1, m_1, n_2 and m_2 are not necessarily integer values.

Multi- cellular model

This flow attempts to create an artificial flow including a range of length scales, each with an associated velocity scale. It is a crude attempt to produce a flow with homogeneous, mean shear free turbulent-like characteristics.

The flow is composed of a number of cell flows generated using the streamfunction

$$\psi = A \sin\left(\frac{n\pi x}{w}\right)\sin\left(\frac{m\pi y}{h}\right)\cos\left(\omega t\right) \qquad (22.3)$$

The user specifies the constants A, n, m and ω for the largest cells, the number of cell sizes in the flow, N, and the ratio of the smallest to largest cell sizes, L_r. The model automatically generates the constants for the other cell sizes based on the length ratio and those for the large cells. The formulae used for the i^{th} cell size (*1* being the smallest) are

$$\begin{aligned} z_i &= \left(L_r\right)^{N-i/N-1} \\ A_i &= A\left(z_i\right)^{4/3} \\ n_i &= \frac{n}{z_i} \\ m_i &= \frac{m}{z_i} \\ \omega_i &= \omega\left(z_i\right)^{-2/3} \end{aligned} \qquad (22.4)$$

The streamfunctions for the N cells are superimposed to produce the flow.

Turbulent shear model

This theoretical model creates a flow with mean shear together with superimposed velocity fluctuations. The velocity field is chosen to ensure that continuity is satisfied.

$$u = U_{mean} + U_{top}\left(\frac{y}{h}\right)^{1/n} + u'\sin\left(\omega t\right)\sin\left(\frac{ky}{h}\right)\exp\left(\frac{-ry}{h}\right) \qquad (22.5)$$

$$v = -v'\sin\left(\omega t\right) \qquad (22.6)$$

Here h is the frame height in mm. U_{mean}, U_{top}, n, u', k, r, v' and ω are constants set by the user.

Sediment motion model

The model creates a flow that is intended to simulate the random motion of sediment particles. The number of particles in each frame that move (the rest remaining stationary), and how far they move in the x and y directions are specified. The movement is a randomly distributed displacement around a mean value with a spread specified that gives the upper and lower bounds of the displacement between two frames.

- *Percent move* – the percentage of the particles in each frame that move.
- *Mean x distance (mm)* – the mean distance a moving particle moves in the x direction.
- *Mean y distance (mm)* – the mean distance a moving particle moves in the y direction.
- *Spread in x distance (mm)* – the spread of distances moved in the x direction (this spread is centred on the mean).
- *Spread in y distance (mm)* – the spread of distances moved in the y direction (this spread is centred on the mean).

Custom formula model

Three theoretical models, one that creates a particle 2D record, one that creates a particle 3D record, and one that creates a particle 3D record where all particles lie on a surface, are based on mathematical formulae provided by the user (see chapter 46 for more information on formulae). Note that in all cases scalar field custom functions created in a tablet can be dragged and dropped into the formulae used in these models - see the User's Guide – Tablet Objects manual. In the case of a particle 2D record

two formulae are given, one for each of the x and y velocity components, while for the particle 3D record 3 velocity components are given. In the case of a particle 3D record with particles on a surface a formula is also given for the initial surface elevation, z0, above the x-y plane. In the first frame all particles will be located on this surface. A range of mathematical functions can be used as well as the independent variables, x, y, possibly z, and t.

Matching

The user can choose to pre-match particles in each frame. This option is useful in testing the performance of a matching-based costing.

The following parameters are set here:

- *Pre-match particles* – if this is selected then particles will be pre-matched for each frame in the record. If this is not selected the remaining three controls are unavailable.
- *Fraction of correct matches* – this text box takes a number between *0* and *1* to indicate the fraction of the particles in the frame that should be pre-matched correctly.
- *Fraction of bad matches* – this text box takes a number between *0* and *1* to indicate the fraction of the particles in the frame that should be pre-matched incorrectly. These incorrect matches will include both bad and false matches.
- *Window for bad matches (mm)* – a particle that is selected as an incorrect match for another particle must lie within a square/cube centred on the particle being matched. This is to simulate the effect of a search window in the proper particle matching process.

Integration

The location of each particle in each frame is obtained by integrating along the path of the particle using a 3rd order Runge-Kutta integration. The number of steps used between frames in this integration can be set on this tab. In addition the user can specify whether the two axes wrap around on themselves. If an axis wraps around particles leaving the flow domain along this axis are reintroduced to the flow on the other side of the domain. For example if the x axis is specified to wrap around a particle travelling in the x direction that leaves the right hand side of the domain will reenter on the left hand side.

The following parameters are set here:

- *Number of Runge-Kutta Steps* – specifies the number of Runge-Kutta steps to use in integrating the particle path between two frames.

- *Wrap x axis* – (not available for processes that create particle 3D records with particles on a surface) if selected the x axis wraps around.
- *Wrap y axis* – (not available for processes that create particle 3D records with particles on a surface) if selected the y axis wraps around.
- *Wrap z axis* – (not available for processes that create particle 2D records or particle 3D records with particles on a surface) if selected the z axis wraps around.

22.4 Particle record processes - other

Currently three other particle record processes are available.

22.4.1 Overlay particle 2D records process

Sometimes the images in an image sequence are interrogated for particles in a number of phases. For example, an initial analysis might identify particles in a particular region of each image because particles in that region had particular characteristics. A second analysis might find particles in the remainder of the image. The particle records obtained from these two processes need to be merged at some point in order to produce a complete particle record.

The overlay particle 2D records process enables the user to take multiple particle records and merge them. It is assumed that all particle records possess the same time step between frames. In merging the records the user can select a temporal offset and spatial x and y offsets if they wish. The temporal offset is always non-negative and is relative to the first record in the list of records to merge (this temporal offset cannot be set for the first record). The offset is specified by the number of frames the first frame in the record is offset from the first frame in the first record. Thus if this offset is set to 0 for all records they will overlay one another. Thus records could be concatenated with a judicious use of the temporal offset.

Spatial offsets are specified in physical dimensions (e.g. mm) and can be positive or negative.

Simplified versions of this process are available by dragging and dropping a particle record onto the particle view of another particle record. See the descriptions of the particle views in chapters 6 and 9 for further discussion.

When creating the process the user is presented with a dialog box that requires the following inputs.

Process

- *Overlay process name* – the name of the overlay process.

Record

- *Overlay particle record name* – the name of the overlaid particle 2D record.

Files

- *List box…* – this list box displays the particle records that are to be overlaid. Each record has the temporal and two spatial offsets listed in brackets following the record file name.
- *Add* – this button allows the user to add particle record files to the list. In addition the user is prompted for the offset. These offsets are displayed in the list box along with the record file name.
- *Insert before* – this button allows the user to insert a record file before the file selected in the list box.
- *Delete* – this button deletes any selected files in the list box.

22.4.2 Create camera 3D calibration image process

The cameras used in three-dimensional particle tracking must be carefully calibrated – as discussed in chapter 10. The calibration involves recording the image of a particle array in 3D space for which the exact particle locations are known. One method for producing these calibration images is through the create camera 3D calibration image process. The concept is simple. The calibration image is a two-dimensional image of particles whose locations are accurately known. The user can then locate this two-dimensional image in 3D space in such a manner that the spatial coordinates of the particles are known. For example the two-dimensional image could be accurately placed against a flow boundary corresponding to $z = 0$.

The user has some control over the layout of the particles in the two-dimensional image. The first option is to locate the particles on a rectangular gird. The second option is to locate the particles in somewhat random manner. While the particles are not fixed to a regular grid they may not entirely randomly located in space. Instead the user has the

option of dividing the image into a rectangular grid of cells and specifying the number of particles per cell. These particles are then randomly located within each cell. This leads to an irregular pattern of particles but one where the particles are reasonably evenly spread across the image.

The particles are drawn with a Gaussian intensity profile (actually an inverted one as the particles are drawn black on white) so that more accurate algorithms can be used for particle identification in *Streams*. The user can specify the radius of the particles and the standard deviation, sigma, used in their Gaussian intensity profile. These two parameters are independent. The particle size is limited by the specific radius and the variation of intensity within this region is determined by sigma.

When executed this process generates three files. The first is the TIFF image file that contains the particle images. The second is an identical image file with the addition that each particle's number is drawn next to it. The third is a text file that provides list of the particles, their physical coordinates (x, y, and z) and their radii.

When creating the process the user is presented with a dialog box that requires the following inputs.

Process

- *Process name* – the name of the calibration image process.

Files

- *Image file name* – the name of the file that the calibration image will be written to. This must be a .tiff or .tif file as tif files provide accurate printing in physically meaning coordinates. Two image files are actually produced. The second has the text "_Num" added to the file name. Both images include the particles but the second image also has the particle numbers drawn next to them.
- *Particle data file name* – the name of the file that the particle information will be written to. The file has the following format. The number of particles in the file is written on the first line. On each subsequent line the information about one particle is written. This information includes the particle number (starting at 0 and corresponding to the numbers in the second image file), the x, y and z coordinates of the particle and its radius.

Parameters

- *Radius (mm)* – the radius of the particles.
- *Sigma (mm)* – the standard deviation of the particle intensity profile.
- *z plane (mm)* – it is assumed that the particles lie in a plane for which the z coordinate is fixed – hence only x and y vary for each particle. This is the value of z that will be output in the particle data file.
- *Width (pix)* – the width of the image in pixels. This parameter needs to be chosen in conjunction with the resolution – see below.
- *Height (pix)* – the height of the image in pixels. This parameter needs to be chosen in conjunction with the resolution – see below.
- *Resolution (dpi)* – this is the resolution of the .tif image in dots or pixels per inch. This parameters should be selected first and then the width and height parameters set based on the physical size of image desired.
- *Use random layout* – when checked the layout uses the pseudo-random layout with particles randomly located within a number of user specified cells. If not checked the particles are located on a rectangular grid. this is the resolution of the .tif image in dots or pixels per inch. This parameters should be selected first and then the width and height parameters set based on the physical size of image desired.

Layout

- *Number of particles/cell* – the number of particles to be randomly distributed within each cell when a random layout is selected.
- *Number of x cells* – the number of cells in the x direction when a random layout is selected.
- *Number of y cells* – the number of cells in the y direction when a random layout is selected.
- *Number of x grid pts* – the number of grid points in the x direction, used when a random layout is not selected. A particle is located at each grid point.
- *Number of y grid pts* – the number of grid points in the y direction, used when a random layout is not selected. A particle is located at each grid point.

22.4.3 Create 3D calibration target process

This process is closely related to that described in section 22.4.2. In order to explore the performance of a particular calibration target (i.e. the distribution of particles in 3D space with known particle coordinates) this process creates a particle 3D record that mimics a calibration target constructed from a number of planes of artificially created particles. Each plane could be constructed in practice using the process in section 22.4.2.

The distribution of the particles in each plane can be specified in the same way as described in section 22.4.2 – namely using cells of randomly distributed particles or regular grids of particles. In addition the user can choose whether each z plane should include the same set of particles (i.e. with the same x and y coordinates but with differing z coordinates) or whether as different set of particles should be generated for each plane.

When creating the process the user is presented with a dialog box that requires the following inputs.

Process

- *Process name* – the name of the process.

Record

- *Record name* – the name of the particle 3D record produced by the process.
- *Width (mm)* – the width of the created record.
- *Height (mm)* – the height of the created record.
- *Depth (mm)* – the depth of the created record.
- *Use random layout* – when checked cells of randomly located particles are used. If not checked the particles will lie on a rectangular grid.

Particle

- *Minimum radius (mm)* – the minimum radius of a particle. The actual radius of a particle is randomly chosen to lie between the minimum and maximum values.
- *Maximum radius (mm)* – the maximum radius of a particle. The actual radius of a particle is randomly chosen to lie between the minimum and maximum values.
- *Minimum intensity (0-255)* – the minimum intensity of a particle. The actual intensity of a particle is randomly chosen to lie between the minimum and maximum values. All particles are monochrome.
- *Maximum intensity (0-255)* – the maximum intensity of a particle. The actual intensity of a particle is randomly chosen to lie between the minimum and maximum values. All particles are monochrome.

Layout

- *Number of particles/cell* – the number of particles to be randomly distributed within each cell in each plane when a random layout is selected.

- *Number of x cells* – the number of cells in the x direction when a random layout is selected.
- *Number of y cells* – the number of cells in the y direction when a random layout is selected.
- *Number of z planes*– the number of planes of particles in the target when a random layout is selected.
- *Δz (mm)* – each plane of particles corresponds to a fixed value of z . The first plane is located in the z = 0 plane and the z coordinate of each subsequent plane is *Δz* larger than the one before. *Use* when a random layout is selected
- *Use same particles* – if this check box is ticked the x-y coordinates of the particles in each plane will be identical – as if one plane of particles is simply moved to different z locations. If not ticked each plane has a different set of randomly located particles. Used when a random layout is selected
- *Number of x grid pts* – the number of grid points in the x direction, used when a random layout is not selected. A particle is located at each grid point.
- *Number of y grid pts* – the number of grid points in the y direction, used when a random layout is not selected. A particle is located at each grid point.
- *Number of z grid pts* – the number of grid points in the z direction, used when a random layout is not selected. A particle is located at each grid point.

22.5 Lagrangian path field processes

Streams provides three Lagrangian path field processes.

22.5.1 Overlay Lagrangian path fields 2D process

Lagrangian path fields are frequently the preferred tool for the analysis of structural testing applications. Such systems invariably have multiple cameras, each generating its own image sequence, particle record and Lagrangian path field. At some point in the analysis process these datasets may need to be combined. The overlay Lagrangian path fields 2D process enables multiple Lagrangian path fields to be overlaid.

The first Lagrangian path field specified acts as an anchor for the process. Other path fields are added to this field. It is assumed that the overlaying of these fields makes sense. In other words, all of the fields possess the same spatial and temporal scales, and the same time step

between frames in the path lines. The user is at liberty to specify x and y offsets (negative or positive) when each of the path fields is incorporated into the overlaid field. This is true even for the anchor path field. In addition the user can specify a time offset, measured in numbers of frames and always positive, so that path lines in the added path field can start later than the first frame in the anchor path field. In this way this process can mimic the concatenation of path fields, if this makes physical sense.

When creating the process the user is presented with a dialog box that requires the following inputs.

Process

- *Overlay process name* – the name of the overlay process.

Record

- *Overlay Lagrangian path field name* – the name of the overlaid Lagrangian path field.

Files

- *List box…* – this list box displays the path fields that are to be overlaid. Each field has the temporal and two spatial offsets listed in brackets following the path field file name.
- *Add* – this button allows the user to add Lagrangian path field files to the list. In addition the user is prompted for the offsets. These offsets are displayed in the list box along with the file name.
- *Insert before* – this button allows the user to insert a Lagrangian path field file before the file selected in the list box.
- *Delete* – this button deletes any selected files in the list box.

22.5.2 Transform from one path field 2D process

In particle tracking systems undesirable camera movement is not uncommon. This movement may be due to external influences such as equipment vibration, and its effects generally require removal from the particle record. To do this, a time dependent series of transforms that transform the particle locations in a particle record, or Lagrangian path field, to the particle locations that would have been recorded by a motionless camera, is required.

The transform from one path field 2D process, and the related transform from two path fields 2D process, described in section 22.5.3, compute such a time series of transforms

The transform from one path field 2D process assumes that a single camera records the movement of a set of reference particles that are assumed to be fixed in space. Thus any movement of these particles is due to the unwanted camera motion. The user extracts a Lagrangian path field that includes only the reference particles from the particle record and analyses this Lagrangian path field using the transform from one path field 2D process.

This process computes a two dimensional transform for each frame in the path field that will map the particle locations in that frame, using all paths present in the field, back onto the corresponding particle locations in the first frame. This transform includes a rotation in the plane together with a translation. It is computed so that the RMS error computed from the differences between the locations of the first frame particles and the locations of the transformed particles from a later frame is minimized. A transform in computed for each frame, thus resulting in a time series of rotations and translations. This series of transforms is stored in a vector 3D field 1D (see the Fields manual) with the x and y translations stored in the first two components of the vector field, and the rotation angle (in radians) stored in the third component.

The resulting transform vector field is added to the *Streams* system and appears in the object list view.

See later sections on the transforms for particle records and Lagrangian path fields to see how these transform vectors can be utilized.

When creating the process the user is presented with a dialog box that requires the following inputs. The details view of the process includes the same parameter settings.

Process

- *Process name* – the name of the transform creation process.
- *Include rotations* – if checked the transforms will include both rotations and translations. If not only translations are allowed.

Field

- *Transform field name* – the name of the transform field.

File

- *Lagrangian 2D path field* – the name of the file that contains the Lagrangian path field to be used in the transform computations. *A Select* button allows the user to choose a file from the file system.

22.5.3 Transform from two path fields 2D process

The transform from two path fields 2D process has much in common with its one path field cousin described in the previous section. It, too, computes a time series of transforms stored in a vector 3D field 1D object. However it computes these transforms in a different fashion. In this case it is assumed that two cameras have recorded a common set of particles, and that the first of these cameras is fixed while the second moves relative to it. Thus this process aims to deduce the movement of the second camera relative to the first, based on the different particle locations in the two Lagrangian path fields.

Two steps are required in this process. The first step pairs paths in the first Lagrangian path field with those in the second. Remember the two path fields are recording the motion of the same set of particles, although it is not essential that the two path fields contain exactly the same number of paths – extraneous paths are ignored. This pairing process is a particle matching exercise with much in common with a typical particle tracking algorithm. The key difference is that only one pair of particle frames are matched (the first from each path field). A simple particle matching tool, that utilizes the standard state based costings common to PTV analysis objects, is provided to undertake this pairing. Note that any paths that do not begin in the first frame, in either Lagrangian path field, are excluded from the analysis.

Once the paired particle paths have been deduced the second step follows a similar process to that described for the one path field process in the previous section. The difference being that transforms are computed that map the particles from a frame in the second path field onto the particles in the same frame in the first path field.

The resulting transform vector field is added to the *Streams* system and appears in the object list view.

When creating the process the user is presented with a dialog box that requires the following inputs.

Process

- *Process name* – the name of the transform creation process.
- *Include rotations* – if checked the transforms will include both rotations and translations. If not only translations are allowed.

Field

- *Transform field name* – the name of the transform field.

File

- *Lagrangian 2D path field 1* – the name of the file that contains the first Lagrangian path field (assumed to be obtained from a fixed camera) to be used in the transform computations. *A Select* button allows the user to choose a file from the file system.
- *Lagrangian 2D path field 2* – the name of the file that contains the second Lagrangian path field (assumed to be obtained from the moving camera) to be used in the transform computations. *A Select* button allows the user to choose a file from the file system.

22.5.3.1 Details view

The details view for the process includes the same three tabs as the dialog box described above. However, the details view includes two additional tabs as described here.

Matcher

- *Particle matcher* – this list box lists the particle matcher that will be used to match the path lines in the two Lagrangian path fields, based on the first frame in each path field. By double clicking the matcher the user can choose the costings and set the parameters for the particle matching process. The costings and parameters available are a subset of those discussed in the chapters on PTV analysis objects and costings.

Statistics

- *Path pair table – at the top of this tab a table of the computed path pairs is listed. The table lists the number of the path pair, the numbers of the corresponding paths from the two Lagrangian path fields*, and the particle locations in the first frame from the two paths. This table provides the user with the opportunity to check whether any of the particle pairings appear to be incorrect. If so then the user can adjust the settings in the particle matcher and recompute the pairings.
- *Statistics fields – at the bottom of this tab is a listed of fields that provide some statistics regarding the computed transforms. Six 1D scalar fields are computed*

when the process is executed. These are: the RMS error in the x and y coordinates of the two path fields without the applied transforms, the total RMS error between the particle locations in the two path field before the transforms have been applied, and the same three statistics after the transforms have been applied. These fields provide the user with a strong sense of how effective the transforms have been in removing the unwanted camera motion, and, thus, also provide reassurance that the original path pairings are accurate.

22.6 Eulerian field processes – combining fields

The primary processes available for Eulerian fields involve combining multiple fields. It is not unusual in experimental systems to have multiple image capture devices whose outputs need to be combined in some way. Alternatively, for experimental systems where very large numbers of images are captured it may not be possible to process all of the images at one time, and later in the analysis the various datasets need to be recombined. In both cases *Streams* provides a mechanism for combining Eulerian fields provided all fields have compatible spatial grids and time steps.

Three combinations are possible. These are:

- *Concatenation* (section 22.6.1)
- *Overlay (section 22.6.2)*
- *Subtraction* (section 22.6.3)

These combinations are available for all four Eulerian field types – velocity 2D, intensity, density and shape fields.

22.6.1 Concatenate Eulerian field process

A concatenated Eulerian field time series is constructed by placing the "frames" of a number of field time series in sequence, hence producing a single, longer time series. For each time series in the sequence the user can specify spatial offsets in the x and y directions. These offsets are in units of grid points so that the grids of the resulting time series still align with one another. These offsets are relative to the origin of the first time series in the sequence.

The user is presented with a dialog box that provides the following options.

Process

- *Process name* – the name of the concatenation process.

Field

- *Concatenated field name* – the name of the concatenated field time series.

Files

- *List box…* – this list box displays the time series that are to be concatenated. The order matches that in the resulting series.
- *Add* – this button allows the user to add field time series files to the list. In addition the user is prompted for the grid offsets. These offsets are displayed in the list box along with the field file.
- *Insert before* – this button allows the user to insert a field file before the file selected in the list box.
- *Delete* – this button deletes any selected files in the list box.

22.6.2 Overlay Eulerian field process

An overlaid field time series is constructed by overlaying the frames in each time series instead of adding them to the end of the previous series, as is the case for the concatenated time series. The user has considerable flexibility here as the grids can be offset in both space and time.

An overlay rule must be specified that informs the process how to calculate the field at a grid point if more than one time series defines a field value at that point. The options are to use the field value from the first frame that defines the field, to use the field value from the last frame, or to average the field values from all frames that define the field at the grid point.

The user is presented with a dialog box that provides the following options.

Process

- *Process name* – the name of the overlay process.

Name

- *Overlaid field name* – the name of the overlaid field time series.

Files

- *List box…* – this list box displays the time series that are to be overlain. The order matches that in the resulting series.
- *Add* – this button allows the user to add field time series files to the list. In addition the user is prompted for the grid and time offsets. The grid offsets are in unit of grid points while the time offset is in units of timesteps. These offsets are displayed in the list box along with the velocity field file.
- *Insert before* – this button allows the user to insert a field file before the file selected in the list box.
- *Delete* – this button deletes any selected files in the list box

Overlay rule

- *Use first frame* – if this radio button is checked the first frame overlay rule is used.
- *Use last frame* – if this radio button is checked the last frame overlay rule is used.
- *Average frames* – if this radio button is checked the field is obtained by averaging all frames that include that grid point.

22.6.3 Subtract Eulerian field process

A subtracted field time series is obtained by subtracting the field values of one time series from the field values of another. The two time series must have the same number of frames and the same grid size. This is a useful tool for comparing the velocity fields obtained from the same particle record using different matching strategies or different interpolation schemes.

The user can choose to have the resulting field values normalised by the magnitude of the field values in one of the two fields.

The user is presented with a dialog box that provides the following options.

Process

- *Process name* – the name of the subtraction process.

Name

- *Subtracted field name* – the name of the subtracted field time series.

Files

- *Field A…* – the file containing the first of the fields. The resulting field is A-B.
- *Field B* – the file containing the second of the fields. The resulting field is A-B.

Normalisation rule

- *No normalisation* – if this radio button is checked no normalisation takes place.
- *Use field A* – if this radio button is checked the difference in field values at each grid point is normalised by the magnitude of the field value in field A at that grid point.
- *Use field B* – if this radio button is checked the difference in field values at each grid point is normalised by the magnitude of the field value in field B at that grid point.

22.7 Eulerian field processes – other

Streams provides one other Eulerian field process.

22.7.1 Create model velocity 3D field process

In addition to producing artificial particle 3D records through the use of simple formula based models *Streams* also provides the option of creating a velocity 3D field time series directly from a formula-based model. This provides a simple means for exploring the systems capabilities without the onerous task of gathering real three-dimensional velocity data or bothering with the additional step of creating a particle 3D record and matching particles in order to create a velocity field.

A spatial and temporal grid is specified by the user and three-dimensional velocities are then computed at each node in the grid based on mathematical formulae provided by the user. It is also possible to exclude certain regions from the resulting velocity field, mimicking the option provided in the process whereby a velocity field is created from a particle 3D record.

The formulae can be produced manually, or they can be created by dragging and dropping scalar field custom functions that have been

created in a tablet – see the User's Guide – Tablet Objects manual. Note however that this drag and drop mechanism is only available in the details view of the process once it has been created, not in the model dialog box used to initially define the process.

On creation a dialog box displayed that requires the data below. The velocity components and their derivatives are specified through mathematical formulae. In each case a popup menu in the text area provides the user with a list of the available symbols.

Process

- *Process name* – the name of the process.

Details

- *Velocity field name* – the name of the velocity 3D field time series created by the process.
- *x origin (mm)* – the x coordinate at the left of the grid.
- *y origin mm)* – the y coordinate at the bottom of the grid.
- *z origin (mm)* – the z coordinate at the back of the grid.
- *Width (mm)* – the width of the grid.
- *Height (mm)* – the height of the grid.
- *Depth (mm)* – the depth of the grid.
- *x grid points* – the number of grid points in the x direction. The spacing between grid points is the width divided by one less than the number of grid points.
- *y grid points* – the number of grid points in the y direction. The spacing between grid points is the height divided by one less than the number of grid points.
- *z grid points* – the number of grid points in the z direction. The spacing between grid points is the depth divided by one less than the number of grid points.
- *t grid points* – the number of grid points in time – effectively the number of frames captured.
- *dt (s)* – the time step between temporal gird points.
- *Store derivatives* – is selected the velocity field derivatives are stored with the field. If not selected then velocity gradients are computed through finite differences based on velocity values on the grid as needed.

u velocity

- *u formula* – the formula to be used to compute the u velocity.
- *∂u/∂x formula* – the formula to be used to compute the x derivative of the u velocity. It is only used if the Store derivatives check box is ticked.

- *∂u/∂y formula* – the formula to be used to compute the y derivative of the u velocity. It is only used if the Store derivatives check box is ticked.
- *∂u/∂z formula* – the formula to be used to compute the z derivative of the u velocity. It is only used if the Store derivatives check box is ticked.

v velocity

- *v formula* – the formula to be used to compute the u velocity.
- *∂v/∂x formula* – the formula to be used to compute the x derivative of the v velocity. It is only used if the Store derivatives check box is ticked.
- *∂v/∂y formula* – the formula to be used to compute the y derivative of the v velocity. It is only used if the Store derivatives check box is ticked.
- *∂v/∂z formula* – the formula to be used to compute the z derivative of the v velocity. It is only used if the Store derivatives check box is ticked.

w velocity

- *w formula* – the formula to be used to compute the u velocity.
- *∂w/∂x formula* – the formula to be used to compute the x derivative of the w velocity. It is only used if the Store derivatives check box is ticked.
- *∂w/∂y formula* – the formula to be used to compute the y derivative of the w velocity. It is only used if the Store derivatives check box is ticked.
- *∂w/∂z formula* – the formula to be used to compute the z derivative of the w velocity. It is only used if the Store derivatives check box is ticked.

Regions

- *Excluded regions* – this list box lists all three-dimensional regions that will be excluded from the velocity field calculation. Grid nodes that lie within any of the listed regions will possess undefined velocities. This list box supports drag and drop of 3D regions.
- *Load* – this button enables the user to load 3D regions from a disk file and add them to the excluded regions list.
- *Delete* – this button will delete any regions selected in the list. The user will be asked to confirm their action.

22.8 Miscellaneous processes

22.8.1 Batch process

The batch process provides a simple means for automating the processing of entities in the background. A list of entities, already stored on disk, is provided by the user. The batch process, when executed, opens each of these files in turn, and executes the entity's process

pipeline. On completion the entity and any entities it's pipeline creates are removed from the system. Thus the output from the pipeline must be saved to disk as part of the pipeline or it will be lost. There is no constraint on the number, or type, of entities that can be processed in this way, although the user should be confident that each pipeline will execute correctly before including that entity in the process. The batch process is intended as an "overnight" activity where processing can continue without user interaction.

The following parameters are set here:

- *Batch process name* – the name of the batch process.
- *Files to process* – this list box displays all files selected by the user. The add, insert before… and delete buttons can be used to select files for the list and manipulate them.

23 Image Filter Pipelines

23.1 Description

This chapter introduces image filter pipelines and their component filters. Filters are at the heart of the image processing system in *Streams* and have been discussed in some detail in the *System Theory and Design* Manual. A filter corresponds to an algorithm that takes the pixel intensity information of an image, transforms it pixel by pixel, and produces a new image based on the transformed pixel data.

Each filter has defined input and output data types. The pixel data of the input image must correspond to the expected input data type, and the pixels of the output image will correspond to the output data type. Currently 4 data types are supported. RGB colour, 10 bit gray-scale, 12 bit gray-scale, and real floating point values.

A filter pipeline comprises a sequence of filters that are applied to the image pixel data in order. When executed a filter pipeline will produce another image sequence comprising the filtered images. As for the original image sequence these images are stored on disk, not retained internally because of memory constraints. For this reason the filter pipeline needs the name for these filtered image files to be provided before it can execute. The exception to this rule is when the output of the filter pipeline feeds into another process. In this case the filtered images are temporary artifacts in the process pipeline and they need not be stored to disk. A most common example is during the creation of an intensity field time series from an image sequence. In most normal circumstances the original images are processed by a filter pipeline before being overlaid onto an Eulerian grid. In this case the filtered images act only as an input into the create intensity field process – they are no value in themselves and wouldn't typically be stored on disk.

23.2 Views

Filter pipelines have two views

- *Details view.*
- *Filter view.*

23.2.1 Details view

Description

The details view displays the name of the process, the name of the new image sequence and the name of the files where the filtered images will be stored. It is important to note that when this process is the final process in a process pipeline it cannot execute unless the name of the new image sequence and the name of the filtered files is defined by the user.

- *Filter pipeline name* – the name the user wishes to use to identify this pipeline.
- *Image sequence name* – the name of the image sequence produced by the process.
- *Filtered image file name* – the name to use to construct the file names for the filtered images. A 5 digit number is appended to the end of the file name (before the extension) so that the image files are distinguished from one another.
- *Browse* – this button allows the user to choose the directory in the file structure in which the images are to be placed and to specify their name.

Access

The filter pipeline details view can be accessed through the pipeline's popup menu.

23.2.2 Filter view

Description

The filter view gives the user control over the filters that are included in the pipeline. It is the default view. The filters currently defined are displayed in the list box at the top of the view. The buttons at the bottom of the view provide the following functions:

- *New* – creates a new filter object and adds it to the pipeline. A dialog box is displayed that prompts the user for the choice of filter (see later).
- *Open* – the user can load a previously saved filter from a disk file with extension *.flt* and add it to the pipeline.
- *Insert before…*– allows the user to create a new filter via a dialog box and insert it before the currently selected filter in the pipeline.
- *Delete* – deletes the currently selected filter in the pipeline. The user is prompted to confirm this action.
- *Clear* – this button clears all filters from the pipeline. The user is prompted to confirm this action.

- *Execute* – if the entity that is associated with this pipeline is an image sequence this button executes the pipeline provided it is properly defined. Each image is filtered through the pipeline and stored in a numbered image file whose name and format (BMP, PNG etc) is specified by the user. The result is a new image sequence that appears in the object list and image sequence list views with a name specified by the user. If the entity is not an image sequence this button is disabled.
- *Update* – pressing this button updates all of the pipeline views and informs the owning entity that the pipeline may have changed.

Access

This view is accessed through the popup menu of the filter pipeline, or by double clicking the object.

23.3 Popup menu

The filter pipeline provides the following options in its popup menu.

- *Save as…* – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock/Unlock* – locks or unlocks the pipeline. A locked pipeline cannot be changed by the user. When locked the name of the pipeline is dimmed.
- *Change name…* – opens a dialog box that enables the user to change the name of the pipeline.
- *Open filter view* – opens the object's filter view. This is the default view.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not an image sequence the execution will fail.
- *Close all views* – closes all of the pipeline's currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

23.4 Filters

The various filters that are available to the user will now be discussed. The user selects a filter from the dialog box that is displayed when the user clicks the new or insert before buttons in the filter view. A filter of the selected type is created and added to, or inserted into, the list. The user can then use the popup menu of the filter object to set its parameters.

23.4.1 Popup menu

All filters have a popup menu. This popup menu has the same form for all filters with a couple of exceptions that will be discussed below.

- *Save as…* – allows the user to save the filter in a disk file. The filter can be reloaded from disk in the filter pipeline filter view. Filter files have a *.flt* extension.
- *Open details view* – opens the details view of the filter in which the user can set the various filter parameters. These parameters will be discussed in the following sections under each particular filter.

Two filters, the Barrel pincushion filter and the Polynomial field calibration filter have additional popup menu items. The Barrel pincushion filter has a single additional item:

- *Open analysis view* – opens the analysis view of the filter. This view typically allows the user to explore the filter, or its effects when applied to an image, in more detail.

While the Polynomial field calibration filter includes two additional items.

- *Open analysis view* – opens the analysis view of the filter. This view typically allows the user to explore the filter, or its effects when applied to an image, in more detail.
- *Calibrate* – this command instructs the filter to perform its calibration. This means generating calibration curves for all of the pixel blocks. This process can take tens of seconds.

23.4.2 Details View

Description

The details view displays the filter's parameters. See the appropriate section below for more information on a particular filter. The user must select either the save or cancel buttons to retain or discard the changes to the filter before closing the details view.

Access

A filter details view can be accessed by double clicking the filter object when it is displayed in the list box in the filter pipeline filter view, or through its popup menu.

23.4.3 Absorption colour filter

The absorption colour filter calculates a real value that is proportional to the optical thickness of a fluid body by determining the attenuation of a particular light component as it passes through the fluid body. The fluid body is normally dyed in some way.

The calculation is given by

$$d(x,y) = -\ln\left(\frac{I(x,y)}{I_0(x,y)}\right) \qquad (23.1)$$

where d(x,y) is the calculated optical thickness (actually proportional to it), I(x,y) is the intensity at a pixel, while I_0(x,y) is a reference intensity at the same pixel. Here, the intensity stands for the intensity of the red, green, blue gun, or the gray scale intensity.

The reference intensity can be obtained in two ways, either using a reference image (see section 23.4.4) or using the light intensity of another light component. The System Theory and Design manual provides more details of this process.

Input data type: RGB

Output data type: Real

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Absorption colour* – this drop down box allows the user to select the colour component that is being attenuated, and the colour that is being used as a reference intensity. All possible combinations are available – e.g. red absorption relative to green, blue absorption relative to red etc. In addition combinations of attenuated colours are possible. Hence blue and green relative to red, etc are also available.

23.4.4 Absorption image filter

This is the second of the absorption filters. This one utilises a reference image to provide the unattenuated reference pixel intensities. The size of the image used must match the size of the images in the image sequence. This is checked by the system.

Input data type: Any

Output data type: Real

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Absorption colour* – this drop down box allows the user to select the colour component that is being attenuated. This can be any component of an RGB signal, a 10 bit or 12 bit gray scale intensity or a real value. In addition it is possible to consider the attenuation of the ratio of the red and green, or red and blue, guns in an RGB image.
- *Image file* – the file that contains the reference image.
- *Browse* – this button allows the user to browse the directory structure in order to locate the reference image.

23.4.5 Amplify filter

An amplify filter scales the intensity at a pixel by a given factor. If the resulting intensity is outside the allowable range for that particular data type then the intensity will be set to the upper or lower limit as appropriate. For example if the red intensity of an RGB image is increased to a value greater than 255 by an amplify filter then the red intensity is set to 255.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image type* – the data type of the filter.
- *Red factor* – the scale factor for the red intensity. Only available when the data type is RGB.
- *Green factor* – the scale factor for the green intensity. Only available when the data type is RGB.
- *Blue factor* – the scale factor for the blue intensity. Only available when the data type is RGB.
- *Gray scale 10 bit factor* – the scale factor for the 10 bit gray scale intensity. Only available when the data type is 10 bit gray scale.
- *Gray scale 12 bit factor* – the scale factor for the 12 bit gray scale intensity. Only available when the data type is 12 bit gray scale.
- *Real factor* – the scale factor for the real value stored at a pixel. Only available when the data type is real value.

23.4.6 Barrel pincushion distortion filter

The barrel pincushion filter provides an algorithm that attempts to remove the lens distortion from an image. Barrel pincushion distortion is the most common type of distortion suffered by poor quality or wide-angle lenses. Details of the transformation can be found in the System Theory and Design manual, and the equation used is repeated here in equation (15.2).

$$\mathbf{r}_s = \left(c_0 + c_1 r + c_2 r^2 + c_3 r^3\right)\mathbf{r}_d \qquad (23.2)$$

where $\mathbf{r}_s$ is the pixel location in the source (distorted image), $\mathbf{r}_d$ is the pixel location in the destination (undistorted image), r is the magnitude of $\mathbf{r}_d$, and the c_i are coefficients chosen by the user. It should be noted that this transformation depends on the location of the origin used. Generally the origin is taken as the physical centre of the image but the user is at liberty to change this.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *x centre (mm)* – the x coordinate of the origin used in the transformation. This will default to the centre of the image, but circumstances may arise where the centre is not the appropriate origin. For example an image sequence that has been trimmed unevenly using the Trimmed transformation would use an origin that is not centred on the trimmed image.
- *y centre (mm)* – the y coordinate of the origin used in the transformation. This will default to the centre of the image, but circumstances may arise where the centre is not the appropriate origin. For example an image sequence that has been trimmed unevenly using the Trimmed transformation would use an origin that is not centred on the trimmed image.
- *Coeff 0* – c0 in equation 23.2. This coefficient provides a general scaling of the image without affecting the distortion.
- *Coeff 1* – c1 in equation 23.2.
- *Coeff 2* – c2 in equation 23.2
- *Coeff 3* – c3 in equation 23.2

23.4.7 Bayer colour filter

This filter takes the pixel data from a Bayer filter based CCD or CMOS camera and converts it into a full 32 bit colour image. This is achieved by interpolating the colour information at each pixel from surrounding pixels.

Input data type: RGB

Output data type: RGB

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Red gain* – the relative gains on the three RGB colour signals must be set in software. Effectively these gains set the white balance of the image and they are best obtained by taking an image of a white background and adjusting the gains until the three colour intensities match over the majority of the image. This field sets the red gain. In other words the red intensity will be amplified by this factor.
- *Green gain* – this field sets the green gain. In other words the green intensity will be amplified by this factor.
- *Blue gain* – this field sets the blue gain. In other words the blue intensity will be amplified by this factor.
- *Bayer coding* – this combo box allows the user to select the orientation of the colour filter. The default filter (standard Bayer coding) assumes a red pixel in the top left corner of the image. The inverted Bayer coding corresponds to a blue pixel in the top left corner.

23.4.8 Convert type filter

The convert type filter converts between data types. It provides a useful link between other filters in the filter pipeline whose types are not compatible. The user has a wide range of conversions from which to choose including the ability to convert from the former 10- and 12-bit gray scale formats to the GSVP standard formats now adopted by *Streams*.

Input data type: User selectable

Output data type: User selectable

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.

- *Conversion* – this drop down list box provides a list of conversions that can be performed by this filter. For example Red to real will convert the red intensity from an RGB image into a real value.

23.4.9 Convolution filter

This filter applies a mathematical convolution to each pixel in the image. The convolution is defined by a matrix containing an odd number of rows and columns that can be thought of as overlying the pixel map, centred on the pixel for which the convolution is to be computed. The pixel values underlying the matrix are multiplied by the overlaying matrix values and then summed in order to compute the convolution.

A number of strategies must be specified by the user. These include a strategy for handling convolutions near the edge of the image when the matrix extends beyond the edge of the image, and strategies for handling underflow and overflow conditions – i.e. when the convolution does not lie within the bounds of the image data type.

The user specifies the size of the convolution matrix and provides the matrix elements. The default convolution matrix is always the identity matrix. The application of such a convolution will leave the pixel map unchanged.

The computation time for this filter is directly related to the square of the matrix size, so large convolution matrices will require consider time to compute.

Input data type: User selectable

Output data type: User selectable

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image type* – this drop down list box can be used to select the input (and hence output) image data type.
- *Matrix size* – the number of rows and columns in the convolution matrix. This number must be odd to ensure symmetry and has a maximum value of 21.
- *Edge strategy* – this drop down box lists the available strategies for applying convolutions to pixels where the matrix extends beyond the edge of the image. These include (a) truncating the matrix (simply ignoring the matrix elements beyond the edge of the image), (b) extending the pixel map beyond the edge, either by wrapping the

image from the opposite edge or reflecting the pixels at the edge as if the edge were a mirror, and (c) setting the convolution to zero.

- *Underflow strategy* – this drop down box lists the strategies that can be adopted to handle underflow – i.e. when the convolution is less than the smallest value possible for the image data type. These include (a) setting the convolution to 0 (effectively the minimum value allowed), and (b) setting the convolution to the negative of itself (thus a convolution of -3 would become 3).
- *Overflow strategy* – this drop down box lists the strategies that can be adopted to handle overflow – i.e. when the convolution is greater than the largest value possible for the image data type. These include (a) setting the convolution to the maximum value allowed, and (b) setting the convolution to zero.
- *Scale factor* – sometimes the convolution matrix contains integer values multiplied but some normalisation factor. In this case it is easier to enter the integer values in the matrix and specify the scale factor separately. This text box contains the scale factor that is applied to the convolution after it has been computed.
- *Matrix* – the matrix is entered on the Matrix tab. Matrix elements are simply entered into the table. If the user changes the matrix size the entered values are retained if possible. If the matrix size is increased in size the matrix elements are retained centred on the new central element, while if the matrix size is decreased matrix elements that fit into the new matrix size are retained.

23.4.10 Discrimination 1 filter

The discrimination 1 filter attempts to accentuate the difference between dark and light patches in an image. It does this by scanning an image, row by row, column by column, or both. For each line of pixels scanned the intensity maxima and minima are identified. This list is then analysed so that any maxima and minima whose intensity difference fails to exceed a user defined threshold are excluded from the list or replaced by an average representative turning point. Finally the intensities of all pixels in the line are recomputed based on the list of maxima and minima. This is done in the following way. The intensity at a maximum is set to the maximum intensity that that type of image allows (for a real image this is set to 1). The intensity at each minimum is set to 0. The pixels between each maximum and minimum are recomputed based on the output scheme chosen by the user. If the output scheme is *graded* a pixel is given an intensity that is a proportion to the maximum intensity. This proportion is computed from its initial intensity relative to the initial intensities of its bracketing maximum and minimum in a linear fashion. If the output scheme is binary a pixel is given an intensity of either 0, if its initial intensity was less than the average of the bracketing maximum and minimum intensities, or the maximum intensity.

Input data type: User selectable

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Based on* – the user can select Red, Green, Blue or RGB Gray for RGB images, GrayScale 10 bit or 12 bit, or Real.
- *Scan direction* – the user can select to scan the images in horizontal rows, vertical columns, or vertical columns followed by horizontal rows. The last of these tends to be most effective.
- *Fluctuation definition* – the fluctuation limit set by the user can either by an absolute value, in the units appropriate for the image, or in relative percentage terms. This percentage is calculated on the range of intensities in the column or row of pixels being processed. Thus for images that have an intensity gradation, say from left to right, the percentage option may be advantageous.
- *Fluctuation limit (abs or %)* – the fluctuation limit, specified either as an absolute value or as a percentage – see Fluctuation definition above.
- *Output scheme* – the user can chose a graded intensity output image or a binary intensity output image, as described in the introduction to the filter.

23.4.11 Discrimination 2 filter

Like the discrimination 1 filter the discrimination 2 filter attempts to accentuate the difference between dark and light patches in an image. It is particularly designed to aid in identifying particles in noisy images. Rows or columns of pixels are scanned. For each row or column an average intensity is computed and then the intensity of each pixel in the row or column is compared to the average. If the pixel intensity deviates by more than a user specified amount (either in absolute terms or as a percentage of the average intensity) that pixel is allocated the maximum intensity possible for the type of image being considered (for an image with real values this maximum value is set to 1). If the pixel intensity does not deviate by this amount the intensity is set to 0. Thus the resulting image will comprise pixels with either the maximum or minimum value. It is worth noting that the deviation is measured in absolute terms so this filter will accentuate objects that are less intense or more intense than the average background intensity.

It is not uncommon in noisy images for the pixels comprising a particle vary in intensity and the discrimination 2 filter may create 0 intensity pixels within a region of otherwise maximum intensity pixels. The user

can specify a maximum gap (in pixels) between maximum intensity pixels that it should treat as noise – i.e. the pixels within this gap should be maximum intensity pixels.

Input data type: User selectable

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Based on* – the user can select Red, Green, Blue or RGB Gray for RGB images, GrayScale 10 bit or 12 bit, or Real.
- *Scan direction* – the user can select to scan the images in horizontal rows or vertical columns.
- *Deviation definition* – the deviation limit set by the user can either by an absolute value, in the units appropriate for the image, or in relative percentage terms. This percentage is based on the average of intensities in the column or row of pixels being processed.
- *Deviation limit (abs or %)* – the deviation limit, specified either as an absolute value or as a percentage – see Deviation definition above.
- *Maximum gap* – the gap in pixels between maximum intensity pixels that the filter should treat as noise. Pixels within such a gap will be set to the maximum intensity.

23.4.12 Eliminate objects filter

The eliminate objects filter removes unwanted objects from an image. The objects are identified as being bright regions on a dark background and all pixels within an identified object are set to zero by the filter. Object identification includes two criteria: pixel intensity and size. The user specifies the threshold intensity that a pixel must meet in order to be considered part of an unwanted object in much the same way as for particle identification. For RGB images the intensity of each of the colour guns must exceed the specified colour threshold. A bright region, identified by finding all of the connected pixels that exceed the specified threshold(s), is eliminated provided the number of pixels comprising the region exceeds the minimum number specified by the user but does not exceed the maximum number.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image type* – the data type of the filter.
- *Red threshold* – the threshold for the red intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is RGB.
- *Green threshold* – the threshold for the green intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is RGB.
- *Blue threshold* – the threshold for the blue intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is RGB.
- *Gray scale 10 bit threshold* – the threshold for the 10 bit gray scale intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is 10 bit gray scale.
- *Gray scale 12 bit threshold* – the threshold for the 12 bit gray scale intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is 12 bit gray scale.
- *Real threshold* – the threshold for the real intensity that must be met or exceeded for a pixel to be identified as part of an object. Only available when the data type is real value.
- *Min pixels* – the minimum number of pixels that must be included in a region of pixels that exceed the specified threshold(s) for it to be identified as an object. If set to -1 there is no minimum number of pixels.
- *Max pixels* – the maximum number of pixels that can be included in a region of pixels that exceed the specified threshold(s) for it to be identified as an object. If set to -1 there is no maximum number of pixels.

23.4.13 Extract filter

The extract filter extracts one of the colours at each pixel and sets the others to zero. The colour may be red, green, blue or gray. In the first three cases the intensities of the other two guns are set to zero, while in the final case all three guns are set to the average of their original intensities.

Input data type: RGB

Output data type: RGB

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.

- *Colour to extract* – this drop down box allows the user to choose the colour they wish to extract from the RGB signal. They can choose, red, green, blue or gray.

23.4.14 FFT filter

The FFT filter provides a simple 2D bandpass filter of an image. The user may specify the lower and upper bounds on the wavelengths that are passed by the filter. It can operate on the red, green or blue guns of an RGB image, the average of the three, or a 10 bit or 12 bit gray scale image or a real value image. The output of the filter can be of two types. The first is the filtered image in which case the output data type matches that of the input image. Alternatively the user can choose to retain the PSD from the FFT (using a linear or logarithmic scale), in which case the output data type is real.

Input data type: Any

Output data type: Same as input, or real if PSD is chosen as output.

Parameters

- *Based on* – the user can specify how the image data is to be used to compute the 2D FFT. For gray scale or real images this is simply the intensity at each pixel. For RBG images any of the gun intensities, or their average, can be used.
- *Smallest x wavelength (mm)* – the smallest wavelength in the *x* direction passed by the filter. If no limit is to be imposed this should be set to -1.
- *Largest x wavelength (mm)* – the largest wavelength in the *x* direction passed by the filter. If no limit is to be imposed this should be set to -1.
- *Smallest y wavelength (mm)* – the smallest wavelength in the *y* direction passed by the filter. If no limit is to be imposed this should be set to -1.
- *Largest y wavelength (mm)* – the largest wavelength in the *y* direction passed by the filter. If no limit is to be imposed this should be set to -1.
- *PSD required* – if selected the power spectral density (PSD) is the output from the filter. If not selected the filtered image is output.
- Use log scale for PSD – if the PSD box is selected this box enables the user to choose a logarithmic, instead of a linear, scale for the PSD.

23.4.15 Identity filter

The identity filter is a "do-nothing" filter. It takes any type of data input and leaves the intensity at each pixel unchanged and of the same type as the input.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.

23.4.16 Invert filter

An inversion filter inverts one or more colours in the original image. Inverting a colour means subtracting its intensity from its maximum value. Thus an invert filter that inverts the green gun in an RGB signal will subtract the green intensity from 255 while the red and blue intensities will be left unaltered. The invert all filter performs the inversion operation on all guns. Gray scale images can be processed in the same way. For 10 bit gray scale the intensities are subtracted from 1023, while for 12 bit gray scale they are subtracted from 4095.

Input data type: Any (except real)

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Colour to invert* – this drop down box allows the user to choose the colour they wish to invert. It can be all, red, green, or blue for an RGB image, or 10 bit gray scale or 12 bit gray scale. Real images cannot be processed by this filter.

23.4.17 Mathematical formula filter

A mathematical formula filter computes the new intensity at each pixel based on a mathematical formula provided by the user. The image data itself can be used in the formula, so for example, for an RGB image the user can include the symbols R, G and B in their formula. In all cases the spatial coordinates x and y and time, t, can be used. See chapter 49 for more details on mathematical formulae

Input data type: Any (as specified by the user)

Output data type: Real values

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Base formula on* – this drop down box allows the user to specify the input image data type.
- *Formula* – this text area is used to enter the formula text. A popup menu in this area provides details of the function, operators, brackets and variables that the user can include in the formula. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.

23.4.18 Overwrite regions filter

The overwrite regions filter provides a mechanism for selectively overwriting the pixel information in an image. The areas of the image that are to be overwritten are defined by a list of regions specified by the user. These regions can be created in the image view of an image sequence.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Filter choice* – the type of image data.
- *Red value* – for an RGB image any pixel that lies within the specified regions will have its red intensity replaced by this value.
- *Green value* – for an RGB image any pixel that lies within the specified regions will have its green intensity replaced by this value.
- *Blue value* – for an RGB image any pixel that lies within the specified regions will have its blue intensity replaced by this value.
- *Gray scale 10 bit value* – for a 10 bit gray-scale image any pixel that lies within the specified regions will have its intensity replaced by this value.
- *Gray scale 12 bit value* – for a 12 bit gray-scale image any pixel that lies within the specified regions will have its intensity replaced by this value.
- *Real value* – for a real value image any pixel that lies within the specified regions will have its intensity replaced by this value.
- *Regions to overwrite* – this list box displays the regions that will be overwritten by the value(s) defined above. These regions can be created in the image view of an image sequence and saved on disk. They are loaded into this list box using the Load button. Selected regions can be deleted by selecting them in the list box and clicking the Delete button. It is important to remember that regions can be of two types – internal and external. In the first case the region is defined as the pixels inside the

geometric shape. In the second case the region is defined as the pixels that lie outside the geometric shape.

23.4.19 Parallax factor filter

The parallax factor filter provides a mechanism for compensating for certain effects of parallax. Parallax arises from the varying angle of light rays entering a camera. If this variation in angle is kept very small the impact of parallax can often be ignored. However, when practical considerations require a wide angle lens to be used the impact of parallax can be significant.

This filter assumes that the light detected by the camera originates in a planar light sheet that is perpendicular to the lens axis. It then computes the increased distance of the light path from this light source to each pixel due to the angle of the light entering the lens, and either multiplies or divides the current pixel value by this factor. The maximum vertical and horizontal angles are required in order to be able to compute the parallax factor. These angles correspond to the angle of the light rays detected at the edges of the image.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Operation choice* – whether the current pixel intensity should be multiplied or divided by the parallax factor.
- *Maximum horizontal angle (degrees)* – the angle between a light ray entering the camera along the lens axis and a light ray detected at the left or right hand side of the image at the same height as the lens axis.
- *Maximum vertical angle (degrees)* – the angle between a light ray entering the camera along the lens axis and a light ray detected at the top or bottom of the image at the same lateral position as the lens axis.

23.4.20 Polynomial field calibration filter

The polynomial field calibration filter (PFCF) is designed to convert light intensities into field concentrations (typically dye concentrations). It does this by generating calibration curves, based on a series of calibration images, which can transform the light intensity at any pixel into a field concentration. The size of the calibration images used must

match the size of the images in the image sequence. This is checked by the system.

The user can decide whether the calibration curves are produced pixel by pixel or whether the calibration data is averaged over some small region within each image in order to obtain a more global calibration curve. In one extreme the user could choose to generate a single calibration curve for the whole image.

The calibration curves are polynomials, of user selectable order. They are based a number of calibration images provided by the user. These calibration images are assumed to correspond to a constant field value. A least squares fit is used for the calculation of the calibration coefficients.

The PFCF can work with any type of image.

Input data type: Any

Output data type: Real values

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Horizontal block size (pix)* – the width of the pixel block that is used to generate a calibration curve. The default size is 1 pixel wide by 1 pixel high. In this case every pixel within the image has its own calibration curve based on the equivalent pixels in the calibration images. If the block is greater than one pixel then the calibration information over the whole block is averaged in order to obtain a single calibration curve for all pixels within the block.
- *Vertical block size (pix)* – the height of the pixel block that is used to generate a calibration curve. See above.
- *Intensity cutoff* – during the calibration process, if the calibration intensity at a particular pixel exceeds the intensity cutoff this pixel intensity does not contribute to the calibration.
- *Base calibration on* – this drop down box specifies the type of pixel data that is expected in the calibration and image sequence images. It can be RGB, gray-scale or real.
- *Calibration images* – this list box displays the calibration images for this filter. The user can add, insert before, delete and clear these images. When an image is added to the list the user specifies both the name of the file containing the image and also the field value corresponding to the intensities in the image.
- *Order of polynomial* – the order of polynomial used for each of the calibration curves. Clearly the number of calibration images must equal or exceed the order of the polynomial in order for the least squares fitting process to work.

Analysis view

The PFCF filter provides an analysis view that enables the user to explore the calibration. The calibration curve for any pixel in the image can be viewed. Similarly the calibration coefficients can be viewed for the whole image.

This view cannot be displayed until the filter has been calibrated. This can be done by selecting the "*Calibrate*" option in the popup menu.

Pixel Fit

This tab allows the user to inspect the calibration curve for any pixel in the calibration images.

- *x pixel* – the column (and x coordinate) of the pixel for which the calibration is required.
- *y pixel* – the row (and y coordinate) of the pixel for which the calibration is required. The rows start at the bottom of the image. Thus pixel (0,0) lies at the bottom left hand corner of the images.
- *Generate fit*…– pressing this button generates the calibration curve for the designated pixel. A field object will appear in the calculated fields list and the user can view this field by double clicking it.

Calibration coefficients

This tab allows the user to generate a two-dimensional field of calibration coefficient values based on a rectangular grid.

- *Coefficient* – the coefficient to be displayed. Note that the last item in this drop down box is the least squares rms error.
- *Origin x coord (mm)* – the *x* coordinate, in mm, of the left most grid point.
- *Origin y coord (mm)* – the *y* coordinate, in mm, of the bottom most grid point.
- *Interpolation pts x spacing (mm)* – the spacing, in mm, between grid points in the *x* direction
- *Interpolation pts y spacing (mm)* – the spacing, in mm, between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.

- *Generate coeff field*…– pressing this button generates the coefficient field. A field object will appear in the calculated fields list and the user can view this field by double clicking it.

Access

This view is accessed through the popup menu of the filter. The filter is displayed in an image sequence filter view.

23.4.21 Remove filter

The remove filter sets the intensity of one of the guns in an RGB image to zero, thus effectively removing that colour from the image. The user can select the gun that is to be eliminated.

Input data type: RGB

Output data type: RGB

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Colour to remove* – the colour to be removed from the image. Red, green and blue are the options.

23.4.22 Remove background filter

The remove background filter attempts to remove the background intensity from an image. Any image which is of uniform intensity will produce an image with zero intensity at every pixel after having this filter applied.

A number of algorithms are available for computing the background intensity. The global algorithm dissects the image into a number of uniformly sized rectangles. The average intensity is computed for each separate rectangle and this average intensity is subtracted from the intensity at each pixel within the rectangle. The local algorithm also calculates an average intensity that is subtracted from the intensity at a pixel, but in this case the average is based on the intensities local to the pixel of interest. This average can be calculated in three ways. The first averages the intensities of the pixels lying within a square region at the centre of which lies the pixel of interest. The second also uses a square centred on the pixel but uses only pixels on its edge in its average

intensity computation. The third averages the pixel intensities of the pixels that lie on a cross centred on the particle.

The user has control over the size of the regions used in both of these algorithms.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image type* – the type of image that is input to this filter.
- *Background type* – specifies whether the local or global algorithm is to be used.
- *Local calculation using* – specifies whether a solid square, hollow square or cross are to be used for the local algorithm.
- *Local extent (pix)* – the size of the square or cross for the local algorithm. The number entered is the half width of the square or cross.
- *Global region width (pix)* – the width of the global rectangles.
- *Global region height (pix)* – the height of the global rectangles.

23.4.23 Scale by maximum in regions filter

The scale by maximum in regions filter is designed to help adjust pixel intensities so that they are more uniform. The filter enables the user to focus on specific regions in the images. The maximum intensity in each region is found and all pixel intensities within each region are scaled so that the maximum intensity becomes the maximum allowable (e.g. 255 for a RGB image) and all others are scaled by the same factor.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Maximum choice* – this combo box provides the options for computing the maximum in each region and hence how the scaling will be computed. The options are: to use each of the three RGB guns separately in an RGB image (All guns - RGB); to use the maximum red intensity to scale all colours in an RGB image (Red – RGB); to use the maximum green intensity to scale all colours in an RGB image (Green – RGB); to use the maximum blue intensity to scale all colours in an RGB

image (Blue – RGB); to use the maximum 10 bit gray intensity to scale all intensities in an 10 bit gray scale image (10 bit gray scale); to use the maximum 12 bit gray intensity to scale all intensities in an 12 bit gray scale image (12 bit gray scale); to use the maximum real value to scale all values In a real image (real value).

- *Set pixels outside region to zero* – if selected all pixels outside the specified regions will have their intensities set to zero. If not selected the pixels retain their intensities.
- *Regions* – this list box displays the regions that are used in the scaling process. These regions can be created in the image view of an image sequence and saved on disk. They are loaded into this list box using the Load button. Selected regions can be deleted by selecting them in the list box and clicking the Delete button. The regions can be manipulated through their popup menus and default views. It is important to remember that regions can be of two types – internal and external. In the first case the region is defined as the pixels inside the geometric shape. In the second case the region is defined as the pixels that lie outside the geometric shape.

23.4.24 Spatial average filter

The spatial average filter effectively smears the pixel information by replacing the intensity at each pixel by an average of its surrounding pixels.

The same domains used for computing the background intensity in the Remove background filter are used here to computer the spatial average intensity at a pixel. These involve using all pixels that lie within a user specified square centred on the pixel, using all the pixels lying on the edges of a square centred on the pixel, or using the pixels in a user specified cross centred on the pixel.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image type* – the type of image that is input to this filter.
- *Average calculation using* – specifies whether a solid square, hollow square or cross are to be used for the local algorithm.
- *Averaging region extent (pix)* – the size of the square or cross for the local algorithm. The number entered is the half width of the square or cross. Thus for a solid square of pixel extent the square will include 11 pixels in each direction – 5 either side of the pixel plus the pixel itself.

23.4.25 Subtract colour filter

A subtract colour filter subtracts the intensity of one of the colour guns, or a combination of their intensities, from all three guns in an RGB image. Thus if the user selects to subtract the blue gun, the intensity of the blue gun is subtracted from the red and green guns and the blue gun's intensity is set to 0. If any of these operations lead to a value less than 0 then 0 is used in its place.

Input data type: RGB

Output data type: RGB

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Filter choice* – the colour to be subtracted. The user can choose red, green, blue, average of red and green, average of red and blue, average or green and blue, or an average of all three colours.

23.4.26 Subtract image filter

The subtract image filter uses a reference image file as part of the subtraction process. Either the intensities of the pixels in the reference image are subtracted from the corresponding pixel intensities in the frames in the image sequence or vice versa.

The size of the image used must match the size of the images in the image sequence. This is checked by the system.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image file* – the file containing the reference image used in the subtraction process. The *Browse* button allows the user to navigate the file system.
- *Subtraction order* – specifies whether the reference image is subtracted from each of the images in the image sequence or vice versa.
- *Image type* – the type of data stored in the images – RGB, 10 bit gray-scale, 12 gray-scale or real values.

23.4.27 Subtract image sequence filter

The subtract image sequence filter uses a reference image sequence to supply the images that are to be subtracted from the image sequence for which the filter is defined. The order of subtraction is specified by the user. The images in the two sequences must have matching image dimensions, but the number of images in each sequence need not match. If the reference image sequence has fewer images then the images in the original image sequence without matching images in the reference sequence are left unaffected by the filter.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Image sequence file* – the file containing the reference image sequence used in the subtraction process. The *Browse* button allows the user to navigate the file system.
- *Subtraction order* – specifies whether the image in the reference sequence is subtracted from the images in the image sequence (Image from frame) or vice versa (Frame from image).
- *Image type* – the type of data stored in the images – RGB, 10 bit gray-scale, 12 gray-scale or real values.

23.4.28 Threshold filter

The threshold filter acts as a tool for removing pixels from an image – ie setting pixel intensities to zero. The criterion for setting a pixel intensity to zero is based on a threshold intensity value at the pixel. If the intensity of the pixel exceeds, or falls below, a given threshold value then the replacement occurs. The threshold can be based on a single RBG component, a gray-scale value or a real value. And in turn the replacement can be applied to a single RGB component, or the gray-scale or real value.

Input data type: Any

Output data type: Same as input

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.

- *Threshold* – the threshold value upon with the replacement criterion is based.
- *Remove intensities* – specifies whether values below or above the threshold are replaced.
- *Base threshold on* – the intensity to which the threshold refers. This can an average of all RGB components, or a single one (ie red, green, blue), or it can be 10 bit gray-scale, 12bit gray-scale or real value.
- *Apply to* – this option specifies which intensities are set to 0 if the threshold condition is met. This can be all guns for an RGB image or a single gun (red, green, or blue) or it can be the 0 bit gray-scale intensity, 12bit gray-scale intensity or real value.

23.4.29 White balance image filter

This filter aims to provide a mechanism for white balancing every pixel in an RGB image. It is assumed that the images to be filtered, and the reference image used by the filter, have already been processed using a Bayer colour filter if they were captured by a Bayer colour camera. The filter works in the following way: The user selects a reference colour gun (red, green or blue) and a reference image. The reference image would typically be an image of an experimental domain before it has been contaminated by dye for example. The filter then computes gains at each pixel for the remaining two colour guns such that the multiplication of these two colour intensities by the computed gains will lead to equal intensities for all three colour guns. Thus if the reference image is processed using this filter every pixel in the filtered image will be gray scale.

Input data type: RGB

Output data type: RGB

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Reference colour* – the chosen reference colour (red, green or blue). The gain for this colour will be one.
- *Reference image file* – the user can choose the reference image using the Browse button. This image is used to compute the gains for the two non-reference colour guns.

24 Particle Filter 2D Pipelines

24.1 Description

This chapter describes particle filter 2D pipelines and their component 2D particle filters. Particle filter pipelines have much in common with the image filter pipelines described in the previous chapter. A particle 2D filter processes a 2D particle record, filtering the particles present in each frame according to the filter's designated algorithm. The result of the filtering process is the elimination of some of the particles in each frame. The pipeline combines one of more of these particle filters. Each frame in the processed record is passed to each filter in the pipeline in the specified order and the resulting particle record is that which is output by the final filter in the pipeline.

The particle filter 2D pipeline provides a mechanism by which the user can selectively delete particles from a particle record. The presence of spurious particles in a particle record is not uncommon, and their presence can have a major impact on the quality of the resulting particle matching process and subsequent field generation. Thus tools for their elimination are important components of the analyst's toolbox. The manual tools available to the user through the record's particle view are often sufficient to eliminate spurious particles that lie outside the domain of interest, but they are inherently limited in scope. The filters available in a particle filter pipeline provide more sophisticated tools that may be employed to eliminate particles that lie within or outside the domain of interest.

24.2 Views

Filter pipelines have two views

- *Details view.*
- *Filter view.*

24.2.1 Details view

Description

The details view displays the name of the process.

- *Filter pipeline name* – the name the user wishes to use to identify this pipeline.

Access

The filter pipeline details view can be accessed through the pipeline's popup menu.

24.2.2 Filter view

Description

The filter view gives the user control over the filters that are included in the pipeline. It is the default view. The filters currently defined are displayed in the list box at the top of the view. The buttons at the bottom of the view provide the following functions:

- *New* – creates a new filter object and adds it to the pipeline. A dialog box is displayed that prompts the user for the choice of filter (see later).
- *Open* – the user can add a previously saved filter from a disk file with extension *.parfil2d* and add it to the pipeline.
- *Insert before*…– allows the user to create a new filter via a dialog box and insert it before the currently selected filter in the pipeline.
- *Delete* – deletes the currently selected filter in the pipeline. The user is prompted to confirm this action.
- *Clear* – this button clears all filters from the pipeline. The user is prompted to confirm this action.
- *Execute* – if the entity that is associated with this pipeline is a particle record 2D this button executes the pipeline provided it is properly defined. Each frame is filtered through the pipeline. Note the pipeline changes the particle record for which it is defined, it does not produce a new record.

Access

This view is accessed through the popup menu of the filter pipeline, or by double clicking the object.

24.3 Popup menu

The filter pipeline provides the following options in its popup menu.

- *Save as*… – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock/Unlock* – locks or unlocks the pipeline. A locked pipeline cannot be changed by the user. When locked the name of the pipeline is dimmed.
- *Change name*… – opens a dialog box that enables the user to change the name of the pipeline.

- *Open filter view* – opens the object's filter view. This is the default view.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record 2D the execution will fail.
- *Close all views* – closes all of the pipeline's currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

24.4 Filters

The filters that are available to the user will now be discussed. The user selects a filter from the dialog box that is displayed when the user clicks the new or insert before buttons in the filter view. A filter of the selected type is created and added to, or inserted into, the list. The user can then use the popup menu of the filter object to set its parameters.

24.4.1 Popup menu

All filters have a popup menu. This popup menu has the same form for all filters that will be discussed below.

- *Save as…* – allows the user to save the filter in a disk file. The filter can be reloaded from disk in the filter pipeline filter view. Filter files have a *.parfil2d* extension.
- *Open details view* – opens the details view of the filter in which the user can set the various filter parameters. These parameters will be discussed in the following sections under each particular filter.

24.4.2 Details View

Description

The details view displays the filter's parameters. See the appropriate section below for more information on a particular filter. The user must select either the save or cancel buttons to retain or discard the changes to the filter before closing the details view.

Access

A filter details view can be accessed by double clicking the filter object when it is displayed in the list box in the filter pipeline filter view, or through its popup menu.

24.4.3 Delete tracked regions

The delete tracked regions filter deletes all particles in a set of regions defined by the user. While this can be achieved in a static sense manually within the particle view of the particle record, the filter enables the regions involved to move through time. The movement of the regions is provided by a vector 2D field 1D that contains the x and y displacements of the regions. This vector field is typically generated from the relative x,y displacement calculator of a Lagrangian 2D path field.

This filter is of use in structural testing applications. Imagine a region on a specimen that, due to its optical characteristics generates a number of spurious particles in the particle identification process. As the testing progresses these particles will move with the material under the changing loading conditions. Their elimination cannot be easily achieved manually in the particle view of the record as each frame would need to be treated individually due to the changing locations of the particles. Instead the displacement of particles in the neighbourhood of the region containing the spurious particles can be generated from a Lagrangian path field, together with the region that contains the spurious particles in the first frame. The application of the filter tracks the region through time using the computed displacements, and deletes the particles within the displaced region at each time step.

Parameters

- *Filter name* – a user specified name for the filter. The default name is the filter type.
- *Displacements* – this table lists the x and y displacements to be used in the filtering process.
- *Field file* – this text box lists the path to the field file containing the x and y displacements in the file system. These displacements are stored in a vector 2D field 1D file. The user can enter the path manually, or more commonly, they will browse the file system and select the required file using the Browse button.
- *Browse…* – this button allows the user to browse the file system and select the displacement file (with .v2d1d file extension).
- *Update displacements* – this button loads the displacements from the selected file and updates the displacements table. The filter stores the displacements not the file name and so until this update process is executed the displacements will not be available to the filter.
- *Clear displacement* – this button clears all of the currently defined displacements.
- *Tracked regions* – the currently defined regions to be tracked are listed in this list box. Regions can be accessed through their popup menu or by

double clicking. Regions can also be added to the list through drag and drop.

- *Load* – the load button on the Regions tab allows the user to load one or more tracked regions from file.
- *Delete* – the delete button on the Regions tab allows the user to selectively delete regions from the list of tracked regions.

25 Particle 2D Identification Processes

25.1 Description

Particle identifiers (PIDs) provide the particle identification subsystem within the *Streams* system. These PIDs all work with two-dimensional images. It is their responsibility to process an image in an image sequence in order to produce a list of 2D particles and their associated physical characteristics (see chapter 4). These objects work with the pixel intensity map of each image, analyzing the intensities in order to identify regions that constitute particles.

A number of different particle identifiers are available, and each has its own set of criteria for determining the existence of a particle in the intensity map. These are described below.

Most particle identifiers assume that the particles in an image correspond to regions of pixel intensity that are greater than the local background. This intensity may be the total intensity, or the intensity of a particular gun in an RGB image. In the case of the single threshold colour PID thresholds for each colour component are used and the user can choose whether all three thresholds must be exceeded or only one for a pixel to be identified as part of a particle. For each particle the identifier computes the area of the particle from the number of pixels comprising the particle, an effective diameter calculated by assuming the particle is circular, and if the user has chosen to do so additional shape information is gathered (see chapter 4).

Currently four particle identifiers are provided.

Particle identifier process parameters are accessible through their details view. Many of the parameters in these windows are common across all identifiers. We will first describe the parameters that are unique to each PID. These are entered on the algorithm tab of the dialog box or window.

25.2 Simple threshold mono PID

The simple threshold monochrome PID uses a single intensity threshold to identify a particle. Once a pixel, whose intensity exceeds the threshold, is found within the pixel map the extent of the particle is determined by finding all pixels connected with the original pixel whose

intensities also exceed the threshold. Further details can be found in the *System Theory and Design* manual.

The user can choose to limit the particles found by the PID by placing a minimum and/or maximum limit on the particle's effective diameter, and/or a maximum limit on the aspect ratio. Particles that do not meet these additional requirements are excluded from the particle list.

The parameters that define the PID are now described.

25.2.1 Algorithm data

Parameters

- *Threshold intensity* – the intensity that all pixels in a particle must exceed.
- *Minimum diameter* – if this parameter is set to -1 then there is no lower limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is less than the specified value are excluded from the particle list.
- *Maximum diameter* – if this parameter is set to -1 then there is no upper limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is greater than the specified value are excluded from the particle list.
- *Maximum aspect ratio* – if this parameter is set to -1 then there is no limit on the particle aspect ratio. If this parameter has a positive value all particles whose aspect ratio is greater than the specified value are excluded from the particle list.
- *Use intensity weighted CoM* – if this box is ticked the center of mass of the particle (i.e. its location) is computed by weighting the location of each pixel within the particle by its intensity.

25.3 Simple threshold colour PID

The simple threshold colour PID behaves in the same manner as its monochrome cousin except that now the three separate colour components of a pixel are used in determining whether a pixel is part of a particle. The user specifies separate thresholds for each colour component and whether a pixel must exceed all three thresholds of only one to be counted as being part of a particle. Once a pixel, that meets these criteria, is found within the pixel map the extent of the particle is determined by finding all pixels connected with the original pixel that

meet these criteria. Further details can be found in the *System Theory and Design* manual.

The user can choose to limit the particles found by the PID by placing a minimum and/or maximum limit on the particle's effective diameter, and/or a maximum limit on the aspect ratio. Particles that do not meet these additional requirements are excluded from the particle list.

The parameters that define the PID are now described.

25.3.1 Algorithm data

Parameters

- *Red threshold* – the red intensity used as a criterion for identifying pixel particles.
- *Green threshold* – the green intensity used as a criterion for identifying pixel particles.
- *Blue threshold* – the blue intensity used as a criterion for identifying pixel particles.
- *Threshold rule* – the user can choose to require only one, or all three, colour thresholds to be exceeded.
- *Minimum diameter* – if this parameter is set to -1 then there is no lower limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is less than the specified value are excluded from the particle list.
- *Maximum diameter* – if this parameter is set to -1 then there is no upper limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is greater than the specified value are excluded from the particle list.
- *Maximum aspect ratio* – if this parameter is set to -1 then there is no limit on the particle aspect ratio. If this parameter has a positive value all particles whose aspect ratio is greater than the specified value are excluded from the particle list.
- *Use intensity weighted CoM* – if this box is ticked the center of mass of the particle (i.e. its location) is computed by weighting the location of each pixel within the particle by its intensity.

25.4 Dual relative threshold PID

The dual relative threshold PID uses a peak intensity threshold to first identify a pixel that is part of a particle. Once such a pixel has been

located, pixels in the neighbourhood are searched in order to find the pixel with the maximum intensity. The threshold intensity for that particular particle is then calculated by multiplying the maximum intensity by an edge threshold factor (which has a value less than or equal to 1). The algorithm now mirrors the single threshold PID by finding all of the pixels connected to the pixel with the maximum intensity whose intensities exceed the threshold intensity. Each particle found with this algorithm will have a different threshold intensity based on its maximum value. Further details can be found in the System Theory and Design manual.

It should be noted that the dual relative threshold PID does not work separately with the colour components of a pixel in the manner of the single threshold colour PID. A single intensity value is used in the search. However, the particle that is found can retain the colour information of the pixels of which it is comprised or it can be saved as a monochrome particle. In either case the colour information is obtained by averaging over the pixels comprising the particle.

The user can choose to limit the particles found by the PID by placing a minimum and/or maximum limit on the particle's effective diameter, and/or a maximum limit on the aspect ratio. Particles that do not meet these additional requirements are excluded from the particle list.

The parameters that define the PID are now described.

25.4.1 Algorithm data

Parameters

- *Peak threshold intensity* – the intensity that first identifies that a particle has been located.
- *Edge threshold factor* – the threshold intensity that determines whether a pixel is part of a particle is calculated by multiplying the maximum intensity by the edge threshold factor.
- *Minimum diameter* – if this parameter is set to -1 then there is no lower limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is less than the specified value are excluded from the particle list.
- *Maximum diameter* – if this parameter is set to -1 then there is no upper limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles

whose diameter is greater than the specified value are excluded from the particle list.

- *Maximum aspect ratio* – if this parameter is set to -1 then there is no limit on the particle aspect ratio. If this parameter has a positive value all particles whose aspect ratio is greater than the specified value are excluded from the particle list.
- *Use colour* – if selected the red, green and blue intensities of the particle are stored. These are averaged over all the pixels comprising the particle. If not selected the particle is saved with the same values for the red, green and blue components of its colour. These are computed by averaging the red, green and blue intensities of all the pixels comprising the particle.

25.5 Dual threshold Gaussian PID

The dual threshold Gaussian PID is designed to work with particle tracking velocimetry (PTV) systems where the particles are small and roughly spherical in shape. For larger particles this algorithm does not perform well.

Two thresholds are used to define a particle. The peak threshold intensity is used to identify a particle, while the edge threshold intensity determines which pixels in the neighbourhood are part of the particle. At least one pixel, whose intensity must exceed the peak threshold, is required before a particle is identified. The two key differences between this algorithm and the previous ones are in the way it handles multiple maxima within a region, and how the particle position is computed.

If more than one intensity maxima is found within the region of pixels whose intensity exceeds the edge threshold, the algorithm assumes that each peak is associated with a separate particle. The other two algorithms treat the peaks as part of the same particle.

This PID tries to obtain a more accurate estimate of the particle position by fitting Gaussian curves to the intensity profiles in the x and y directions. Provided there are at least three pixels within the particle in each of these directions this fitting process provides sub-pixel accuracy for the particle position. If less than three pixels are available a simpler algorithm is required. If only as single pixel with an intensity exceeding the edge threshold exists in one direction then the particle location is at the centre of that pixel. If two pixels in that direction have intensities greater than the edge threshold intensity then the particle location is dependent on the intensity of the two pixels. If their intensities are the same the particle is located on the boundary between them. If the

intensity of the lesser pixel equals the edge threshold intensity the particle is located at the centre of the pixel of maximum intensity. For intensities lying between these two extremes the particle location moves between these two locations in a linear dependence on the intensity difference. . Further details can be found in the System Theory and Design manual.

The user can choose to limit the particles found by the PID by placing a minimum and/or maximum limit on the particle's effective diameter, and/or a maximum limit on the aspect ratio. Particles that do not meet these additional requirements are excluded from the particle list.

The parameters that define the PID are now described.

25.5.1 Algorithm data

Parameters

- *Peak threshold intensity* – the intensity that first identifies that a particle has been located.
- *Edge threshold intensity* – all pixels whose intensity exceeds the edge threshold intensity will be included in the particle.
- *Minimum diameter* – if this parameter is set to -1 then there is no lower limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is less than the specified value are excluded from the particle list.
- *Maximum diameter* – if this parameter is set to -1 then there is no upper limit on the particle diameter (measured in the same units as the conversion factors specified when the image sequence was created – assumed to be mm). If this parameter has a positive value all particles whose diameter is greater than the specified value are excluded from the particle list.
- *Maximum aspect ratio* – if this parameter is set to -1 then there is no limit on the particle aspect ratio. If this parameter has a positive value all particles whose aspect ratio is greater than the specified value are excluded from the particle list.
- *Base intensity on* – fro RGB images the user can select how the intensity is to be calculated. It can be based on an average of all three colours or based on just one of the guns.
- *Use colour* – if selected the red, green and blue intensities of the particle are stored. These are obtained from the colour of the pixel that has the maximum intensity. If not selected the particle is saved as a monochrome particle. The intensity is the average of the maximum intensities predicted by the two Gaussian fits to the intensity profiles. If this exceeds 255 then the intensity is set to this value.

25.6 Common parameters

As mentioned above all PIDs have some data in common. This is described here based on the tabs on which the parameters are entered

Process

- *Process name* – the name of the particle identification process.

Record

- *Particle record name* – the name of the particle record that is output by the process.

Definitions

- *Base intensity on* – this combo box allows the user to choose how the intensity is to be computed at each pixel.
- *Store shape information* – if this is selected the additional particle shape information is stored with the particle. This information is the moments of inertia about the principal axes (this gives the aspect ratio), the angle of the major principal axis to the x axis (theta) and the eccentricity.
- *Store particle intensity matrix* – if this is selected the intensity matrix, the rectangular matrix of pixels that contains the particle, is stored with it. This is memory intensive and should be avoided if possible.

Frames

- *All frames* – if selected all frames in the image sequence are processed.
- *Collapse frames* – if this is selected all particle information is stored in a single frame.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all 2D regions that the user wishes to exclude from the particle identification process. This might include regions of reflected light. These regions are most easily created in the image sequence image view. Regions in the list can be manipulated and viewed through double and right clicking on a region.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

25.7 Popup menu

The particle identification process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not an image sequence he execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

26 PTV 2D Analysis Pipelines

26.1 Description

A PTV 2D analysis pipeline is a bound process that supports the particle matching optimization system for particle 2D records described in the *System Theory and Design* manual. The pipeline comprises a number of PTV 2D analysis objects that implement the matching process. Each analysis employs one or more costing strategies for determining the costs to be used in the optimization process. The PTV 2D analyses are executed sequentially so that the particle matches produced by the first analysis will be used as the basis for the matches produced by the second and so on.

When a PTV 2D analysis pipeline is first created it is given a default name and appears in the process view of the entity to which it is attached. Changes to the pipeline, including the creation of analysis objects and their associated parameters, are achieved through the pipeline's various views, accessed via its popup menu.

26.2 Views

The PTV 2D analysis pipeline object has two views.

- *Details view.*
- *Analysis view.*

26.2.1 Details view

Description

The details view displays the name of the process and enables the user to change it.

Access

The PTV 2D analysis pipeline details view can be accessed through the pipeline's popup menu.

26.2.2 Analysis view

Description

The analysis view gives the user control over the PTV 2D analyses that are included in the pipeline. It is the default view. The analyses currently defined, and their statuses (WAITING, ABORTED or EXECUTED), are displayed in the list box at the top of the view. The status of a PTV analysis simply indicates to the user whether the current analysis has been executed. This status does not restrict the user's actions. For example a completed analysis can be executed again without resetting it first. Analyses can be edited after they have executed and so on. The buttons at the bottom of the view provide the following functions:

- *New* – creates a new PTV 2D analysis object and adds it to the list.
- *Load* – loads a previously saved PTV 2D analysis object from disk and adds it to the list. These have a file extension of *.ptv* or *.ptv2*.
- *Delete* – deletes a selected PTV 2D analysis object from the list. The user is prompted to confirm this action.
- *Reset* – the *Reset* button resets all analyses to a WAITING status.
- *Clear* – this button clears all analyses from the list. The user is prompted to confirm this action.
- *Delete matches* – if the pipeline is associated with a particle record this button deletes all matches currently defined for the record. The user is prompted to confirm this action. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.
- *Execute* – if the pipeline is associated with a particle record this button executes all analyses sequentially. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the PTV 2D analysis pipeline, or by double clicking the object.

26.3 Popup menu

The PTV 2D analysis pipeline provides the following options in its popup menu.

- *Save as…* – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the pipeline.

- *Change name…* – opens a dialog box that enables the user to change the name of the pipeline.
- *Open analysis view* – opens the object's analysis view.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the pipelines currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

26.4 PTV 2D analysis

A PTV 2D analysis object can be created in the PTV 2D analysis pipeline analysis view by pressing the New button. The new analysis is added to the pipeline and the user changes its settings, including creating costings, through its details view.

26.4.1 Details view

The details view includes four tabs, and are discussed in the following sections.

Analysis

The analysis tab allows the user to specify a name for the analysis being defined. This name is generated automatically by *Streams* whenever the user changes the selected costings. The automatic name includes the selected costings, and the process type. The user should only change this name once the process type and costings have been chosen.

Four types of process can be performed by a PTV 2D analysis object. The first is a global optimisation process by which particles in one frame are matched to those in the next, or previous, frame, in an optimal way such that the total cost of the particle matches is minimised. This is the default process. The second process is referred to as a local optimisation. In this case matches are made between particles only when the match is the optimal match for the particles in both frames. In other words, when all possible matches for a particle in frame 1 are considered the only possible match is to the particle in frame 2 that carries the lowest cost. That match will only be made if no other particle in frame 1 can be matched to the frame 2 particle with a lower cost. This process will generally lead to fewer matches than the global optimisation but it should, provided appropriate costings have been

chosen, generate better quality matches. The third process is a residual optimization in which only the currently unmatched particles participate. In this process the current matches are left unaffected. Finally it is also possible for the PTV analysis to perform a cleanup process. A cleanup uses the defined costing strategies to break matches that are currently defined if their cost exceeds the MMC. No new matches are created. The cleanup is particularly useful in removing incorrect matches made by a previous analysis. A number of the parameters defined for the PTV 2D analysis object are irrelevant when a cleanup is performed - for example the search window (although the distance costing does use this parameter).

Costings

The costings tab allows the user to choose the 2D costing strategies that the PTV object will employ in calculating particle matching costs. The user can select any number of costing strategies. Each type of costing has a number of parameters (see below) that are specified by the user in its details view. This tab has the following elements:

- *Costings defined for this analysis* – this list box lists all of the costings currently defined for the analysis. Costings can be accessed through their popup menus. This list box supports drag and drop for costings.
- *New* – presents a dialog box from which the user can select a new costing. The costing is added to the list box (see above) and the user can change the costing parameters through its details view.
- *Load* – loads a previous saved costing from disk. Costings have a *.ctg* extension.
- *Delete* – deletes all costings currently selected in the list. The user is asked to confirm this action.
- *Clear* – clears all costings from the list. The user is asked to confirm this action.

Optimisation

The majority of the parameters governing the particle matching process are set in the optimisation tab of the PTV 2D analysis details view.

The following parameters are set here:

- *MMC* – this is the maximum matching cost. If the cost of matching a particle in frame 1 to a particle in frame 2 exceeds this value the match is not considered in the optimisation process.
- *Search window* – the search window defines a rectangular window, positioned relative to a particle in frame 1. Only particles lying within this window in frame 2 are considered as possible candidate matches for the frame 1 particle. This window allows the user to severely restrict the

number of particles that are considered as possible matches to any particle in frame 1, thus vastly reducing the processing time required to perform the optimisation. The window coordinates are in mm. Note the window is defined by the position of its bottom, left corner relative to p1 (negative values mean to the left of, and below, p1), and its width and height, and its definition is independent of the direction of iteration through the frames. By default the search window is centred on the particle and has a width and height which are 10% of the record's window width and height. It is also possible for the user to choose to exclude a circular region, centred on the particle, from the search window. The radius of the excluded region (in mm) is specified in the Exclusion radius parameter. This feature can be useful in eliminating stationary particles from the matching process.

- *Ensure forward and backward matches agree* – by checking this box the user restricts the matches only to those that agree if the frames are analysed in both directions – is matching particles in frame 1 to those in frame 2 and then matches those in frame 2 to those in frame 1. For many PTV analyses these two matching processes will yield the same results (eg with a search window centred on the particle and using a state based costing such as distance costing) and hence checking this box will have no effect.
- *Number of iterations* – while the problem of matching particles in one frame to those in the next is a linear programming problem, the problem of matching particles throughout all frames is not. This is due to the fact that some of the costing strategies use matches from previous, or subsequent frames, in the calculation of their costs. Therefore changes to the matches in one frame can affect the desirability of matches in another. The optimisation process, that works through the frames in the record in sequence making matches according to the chosen costings etc, can iterate through the frames as many times as the user wishes. This parameter specifies the number of iterations.
- *Alternating forward/backward iterations* – if more than one iteration through the frames is performed the user can select whether each iteration is in the same direction through the frames – eg starting at frame *0* and finishing at frame *N-1* every time – or whether the direction should alternate.
- *Initial direction* – the user can choose whether the optimisation process starts at frame *0* and progresses forward through the frames, or starts at frame *N-1* and progresses backward through the frames.
- *Maximum number of sweeps* – it is possible that the auction algorithm does not converge for a particular set of strategies when processing a certain frame. In this case the maximum number of sweeps through the auction algorithm (see *System Theory and Design* Manual) is specified here.
- *All frames* – when the pipeline is associated with an entity other than a particle record the number of frames in the record will not be known ahead of time. This check box tells the system to process all of the available frames.
- *First frame / last frame* – the PTV analysis can be restricted to a subset of the frames in the record. Frames outside this range are not considered in the analysis.

Regions

On occasions it can be desirable to limit the particles that are analysed within a frame. Such a desire can arise because the flow possesses regions with substantially different characteristics that require different matching strategies. The regions tab enables the user to limit the particles involved in the matching process to user specified regions. Regions for both the first and second frame in the matching process can be set. Only particles that lie within these regions will participate in the matching process.

The controls in this tab are

- *First frame regions* – this list box displays the regions used to limit the particles in the first frame involved in the matching process. If this list box is empty all particles participate. If multiple regions are listed then particles lying within any of them will be matched if possible. This list box supports drag and drop of 2D regions.
- *Load* – this button allows the user to load regions from a disk file into the *first frames regions* list box. These regions are typically created within the particle 2D record particle view (see section 6.2.2).
- *Delete* – all selected regions in the *first frame regions* list box are deleted when this button is pressed.
- *Second frame regions* – this list box displays the regions used to limit the particles in the second frame involved in the matching process. If this list box is empty all particles participate. If multiple regions are listed then particles lying within any of them will be matched if possible. This list box supports drag and drop of 2D regions.
- *Load* – this button allows the user to load regions from a disk file into the *second frames regions* list box. These regions are typically created within particle 2D record particle view (see section 6.2.2).
- *Delete* – all selected regions in the *second frame regions* list box are deleted when this button is pressed.

26.4.2 Popup menu

The PTV 2D analysis object provides the following options in its popup menu.

- *Save as…* – saves the analysis object in a disk file. All analysis object files have the extension *.ptv2*.
- *Set status* – changes the status of the analysis object.
- *Open details view* – opens the details view so that the user can change the parametes described in the tabs above.

26.5 Costings in 2D

This section summarises the various costing strategies that are available when a PTV 2D analysis object attempts to match particles. The costings available in *Streams* have been discussed at length in the *System Theory and Design* Manual. We will not repeat these discussions here but would refer the reader to that document.

In the sections that follow, the following terminology will be used. Matches are being sought between two frames, frame 1 and frame 2. P1 is an arbitrary particle in the first frame of the pair being analysed, and a match with an arbitrary particle in the second frame, p2, is being considered.

All costings have a single details view. They do not have a popup menu.

26.5.1 Details View

Description

The details view displays the costing's parameters. All costing details views have two tabs, one which is common to all and the other which is particular to each costing. The common tab includes the following:

Name

- *Name* – the user specified name of the costing. The default is the costing type.
- *Weighting* – the cost is multiplied by this weighting to obtain the total contribution this costing makes to the total cost in a PTV 2D analysis. This parameter is particularly useful when more than one costing is defined in a PTV 2D analysis.
- *Use as default only* – if ticked this costing won't be used unless no other costing is available.

For each costing described below the parameters, peculiar to that costing will be described.

Access

A costing details view can be accessed through the settings button on the Costing tab of the PTV 2D analysis details view.

26.5.2 Adjacency costing

The adjacency costing is a state-based costing that relies on matching particle patterns between frames. Particles that lie within a rectangular region centred on p1 are displaced the same amount as p1 as it moves to p2, and compared with the particles lying within the same rectangular region centred on p2. The distances between the displaced frame 1 particles and their closest neighbours (adjacent particles) in frame 2 comprise the cost.

The parameters set in the view for this costing are:

- *Window width (mm)* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height (mm)* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Max distance (mm)* – if the closest particle in the next frame is further away than this distance then the distance does not contribute to the cost.
- *Default cost* – if this cost can't be calculated, because there are no adjacent particles, the cost is defined to be this value.
- *Use radius norm* – normally the distance between p1 and p2 is used to normalize the cost. If this checkbox is selected the sum of the two radii of the adjacent particles is used to normalise the cost.
- *Only one match* – sometimes two particles from frame 1 will have the same adjacent particle in frame 2. If this check box is ticked only one of the frame 1 particles can have the frame 2 particle as its adjacent particle. Otherwise it can be adjacent to both.
- *Use max distance as default* – if a frame 1 particle has no adjacent particle in frame 2 then the costing can treat this particle in two ways. If this checkbox is ticked then the maximum distance is used as the default distance to the adjacent particle and this distance is included in the costing. If this checkbox is not selected then the particle is ignored and does not contribute to the costing.
- *Use weights* – if this check box is ticked the contributions of each frame 1 particle to the total cost will be weighted by the inverse of its distance from p1. Thus closer particles carry more weight in the determination of the total cost.

26.5.3 Centre of mass costing

The centre of mass costing is a state-based costing that uses the distribution of particles in each frame as a measure of the cost. The location of the centre of mass of the particles that lie within a rectangular

region centred on p1 is compared with the position of the centre of mass of particles within the same rectangular region centred on p2. The size of the particles is not considered in the centre of mass calculation. The cost is the distance between the centre of mass in frame 1, incremented by the displacement between p1 and p2, and the centre of mass in frame2. It is normalised by the distance between p1 and p2.

The parameters set in the view for this costing are:

- *Window width (mm)* – the width of the window that defines which particles are used in the centre of mass calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height (mm)* – the height of the window that defines which particles are used in the centre of mass calculation in each frame. By default the window height is 10% of the record window's height.
- *Default cost* – if this cost can't be calculated because there are no particles within the window in frame 1 or frame 2 then this cost value is used as the default.

26.5.4 Colour costing

The colour costing is a state-based costing that uses the difference in particle colour as a measure of the cost. Each colour component can be weighted so that it contributes more or less to the cost (see the *System Theory and Design* manual).

The parameters set in the view for this costing are:

- *Red weight* – the absolute difference between the red intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.
- *Green weight* – the absolute difference between the green intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.
- *Blue weight* – the absolute difference between the blue intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.

26.5.5 Correlation costing

The correlation costing is a state-based costing. Like the adjacency costing it too is based on comparing the particle patterns within two

rectangles centred on p1 in frame 1 and p2 in frame 2. The cross-correlation of the intensity fields in these two rectangles, normalised by their auto correlations, is used in the cost. However this cross-correlation calculation differs from that used in PIV in two ways. Firstly, the particles themselves do not contribute to its calculation, nor the calculation of the autocorrelations. Secondly, because the pixel fields used in the correlation calculations are artificially created (ie they are not from the original images) the user can perform certain modifications to these fields. In particular the user can magnify or shrink the particles within the rectangles.

The user must provide the following parameters in the details view.

- *Radius factor* – the factor by which the radius is multiplied before a particle is drawn in the artificial correlation window (default value is *1*). The user must ensure that no particle in the flow has a radius (after multiplication by the radius factor) greater than the smaller of half the correlation window width and height.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no cost is available.
- *Correlation window size: Width (mm)* – the width of the correlation window in mm. By default the window width is 10% of the record window's width.
- *Correlation window size: Height (mm)* – the height of the correlation window in mm. By default the window height is 10% of the record window's height.
- *x scale factor (mm/pix)* – the x scale factor that converts mm to pixels. This is required in the construction of the artificial pixel fields.
- *y scale factor (mm/pix)* – the y scale factor that converts mm to pixels. This is required in the construction of the artificial pixel fields.
- *Use Gaussian intensity profile* – if this check box is selected then each particle will be drawn in the correlation window with a Gaussian intensity profile. Otherwise it will be drawn with a uniform intensity.
- *Use partial pixel shifts* – if this check box is selected then the artificial correlation window can be overlaid on frame 2 with sub-pixel accuracy, reflecting the fact that the centres of particles p1 and p2 may not lie in the centre of a pixel.

26.5.6 Distance costing

The distance costing is a state-based costing, depending on the distance of p2 from a predicted position of p1. By default the predicted position of p1 is simply its original position in frame 1, but the distance costing now allows the user to describe the velocity field in which the particle is moving with mathematical formulae, thus providing a rough estimate of

its likely new position. The user also has the choice of biasing the distance to have a heavier dependency on the discrepancy in the x displacement or the y displacement

The user must provide the following parameters in the details view:

- *Power* – the power to which the normalized distance is raised in computing the cost. It must be positive.
- *Y distance weight* – the weight given to the y distance error in computing the distance. If greater than 1 the y error has a greater influence than the x error.
- *Reference length* – the cost must be normalized relative to some reference length. The user sets that reference length in this text box. Typically it would be similar in size to the search window but not necessarily so.
- *x velocity formula* – a formula that predicts the x velocity of the flow field. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.
- *y velocity formula* – a formula that predicts the x velocity of the flow field. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.

26.5.7 Intensity costing

The intensity costing is a state-based costing. The cost is proportional to the difference in intensity of the two particles.

26.5.8 Pseudo-correlation costing

The pseudo-correlation costing has much in common with the adjacency costing but tries to mimic the correlation costing in the manner in which it computes its cost. Like the adjacency costing adjacent particles are found in the two frames. The cost in the pseudo-correlation costing is the average of costs of the adjacent particle pairs. These costs are computed as the distance between the particle centres normalized by the sum of the radii of the particles. If the distance between the centres exceeds the sum of the radii (i.e. the particles do not touch) the distance is set to the sum of the radii. Thus the maximum cost is 1 and minimum is 0, consistent with the correlation costing.

This costing also provides the possibility of increasing or shrinking the radii of the particles in the same way as the correlation costing.

The parameters set in the details view for this costing are:

- *Window width (mm)* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height (mm)* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Max distance (mm)* – if the closest particle in the next frame is further away than this distance then the distance does not contribute to the cost.
- *Default cost* – if this cost can't be calculated, because there are no adjacent particles, the cost is defined to be this value.
- *Radius factor* – the factor by which all particle radii are multiplied before the cost is calculated.
- *Only one match* – sometimes two particles from frame 1 will have the same adjacent particle in frame 2. If this check box is ticked only one of the frame 1 particles can have the frame 2 particle as its adjacent particle. Otherwise it can be adjacent to both.
- *Use weights* – if this check box is ticked the contributions of each frame 1 particle to the total cost will be weighted by the inverse of its distance from p1. Thus closer particles carry more weight in the determination of the total cost.

26.5.9 Shape costing

The shape costing is a state-based costing. The cost can be computed in two ways. In each case it is based on the normalized change in the four shape parameters – diameter, aspect ratio, theta and eccentricity. The first method is to take a weighted sum of the normalized changes in shape based on the specified weights. The second is to use a mathematical formula that includes these change variables.

The user must provide the following parameters in the details view:

- *Diameter weight* – the weight for the normalised change in particle diameter between p1 and p2.
- *Aspect ratio weight* – the weight for the normalised change in particle aspect ratio between p1 and p2.
- *Theta weight* – the weight for the normalised change in theta (angle of the major principal axis) between p1 and p2.
- *Eccentricity weight* – the weight for the normalised change in particle eccentricity between p1 and p2.
- *Use formula* – if selected the cost is computed from the costing formula not the sum of the individual shape parameters.

- *Costing formula* – the user can enter a mathematical formula for computing the cost based on the various normalized shape change parameters.

26.5.10 Direction velocity costing

The direction velocity costing is a matching-based costing. It uses the match to p1 in the previous frame and perhaps the match to p2 in the frame after frame 2 to compute the change in direction of the particle trajectory. Specifically it measures the change in angle of the trajectory if p1 is matched to p2. This angle is normalized by a reference angle specified by the user to compute the cost. Thus, if the particle suffers no change in direction, independent of the change in velocity magnitude, then the cost is zero.

The user can choose whether to use both the match prior to frame 1 as well as the match subsequent to frame 2 to estimate the change in angle in the trajectory, or just use the match prior to frame 1.

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no direction change estimate is available.
- *Norm angle (deg)* – this angle is used to normalize the computed angle to produce a cost.
- *Use before and after* – if this box is ticked the angle measuring the change in direction is the average of that between the displacement from the particle matched to p1 in the previous frame to p1 and the displacement from p1 to p2, and that between p1 and p2 and the displacement from p2 to the particle it is matched to in the subsequent frame. This assumes both matches exist. If not the angle will be based on either one or the other of the changes in direction. If this box is not checked only the first of these two angles is used.

26.5.11 Least squares velocity costing

The least squares velocity costing is a matching-based costing. It uses the velocity estimates at neighbouring particles to calculate a two-dimensional polynomial fit to each of the two velocity components in the region near p1. These two-dimensional polynomials can be of any order from 0 to 4 and are based on a least squares fit to the neighbouring velocities. These polynomials are used to estimate the velocity at the location of p1 and hence to predict the likely displacement of p1 between

frames 1 and 2. The cost is proportional to the distance between p2 and the predicted location of p1.

While this costing is nominally a velocity based costing, actually it is particle displacements that are used in the cost calculation. The cost is normalised by the predicted displacement of p1. For regions of very low velocity this might be very small with the result that the costs are amplified in an undesirable way. This costing, like many of the velocity based costings, allows the user to nominate a default normalisation length. If the displacement of p1 is less than this value the default length is used.

The user must set the following parameters in the details view:

- *Polynomial order* – the order of the two-dimensional polynomials used to fit the velocity components in the region near p1.
- *Number of neighbours* – the number of neighbouring particles used in the calculation of the polynomial fit to the local velocity field. These particles are the nearest neighbours to p1.
- *Maximum distance* – any nearest neighbours that are further from p1 than this distance will be excluded from the least squares polynomial fit to the velocity field.
- *Minimum cost weight* – each velocity used in calculating the velocity polynomial fits are multiplied by a weight factor. This parameter is the weight factor used for the velocity of the particle furthest from p1. The value must lie between 0 and 1. The weight factor for other velocities decreases linearly from 1, for a particle at p1, to the minimum cost weight at the most distant particle.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length(mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.

26.5.12 Local velocity costing

The local velocity costing is a matching-based costing. It uses a local estimate of the velocity in the neighbourhood of p1 to predict the location of p1 in frame 2.The cost is proportional to the distance between p2 and the predicted location of p1.

The local estimate of the velocity for p1 can be obtained in two ways. In the first instance the user can specify a rectangle, centred on p1, and the velocity is based on the current matches of all particles within this rectangle in frame 1 that are currently matched. Either a weighed

average of the velocities is used (the weight varies linearly from 1 at the location of p1 to 0 at the corners of the specified rectangle) or their median value. The second of these is likely to be more accurate if some of the velocities within the rectangle are incorrect. In the second instance the user can choose to use the matches of a number of nearest neighbours to p1 in frame 1. In choosing this option the user specifies a rectangle that all of the nearest neighbor particles must lie within. Again the velocity can be computed using a median or weighted average.

The nearest neighbor option can be particularly useful if the particle density within the image is not uniform. Using a fixed rectangle that is suitable for densely seeded regions of the flow is likely to lead to no velocity estimate at all in sparely seeded regions. The nearest neighbor option ensures that a velocity estimate will be available in almost all cases while at the same time providing some limit on the particles used in obtaining this estimate.

The local velocity costing uses a default normalization length in the same way the least squares velocity costing (see section 26.5.11).

The user must set the following parameters in the details view:

- *Window width (mm)* – the width of the window that defines which particles are used in the calculation of the velocity estimate in each frame, or in the case of the nearest neighbours option, the width of the window the provides an outer bound for the nearest neighbours. By default the window width is 10% of the record window's width.
- *Window height (mm)* – the height of the window that defines which particles are used in the calculation of the velocity estimate in each frame, or in the case of the nearest neighbours option, the height of the window the provides an outer bound for the nearest neighbours. By default the window height is 10% of the record window's height.
- *Use nearest neighbours* – select this check box if the nearest neighbours option is to be used.
- *Number of neighbours* – the number of nearest neighbours used to computer the velocity estimate. Only relevant when the Use nearest neighbours check box is selected. Note that fewer nearest neighbours will be used if there are insufficient particles in the specified bounding window.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.
- *Use weighted average* – if this radio button is checked then a weighted average is used to estimate the velocity.

- *Use median* – if this radio button is checked then the median is used to estimate the velocity.

26.5.13 Magnitude velocity costing

The magnitude velocity costing is a matching-based costing that uses the change in velocity magnitude as the basis for the cost. The magnitude of the velocity based on matching p1 to p2 is compared to a second velocity magnitude using either a previous match for p1 or a subsequent match for p2 (or both is they are available).

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 26.5.11 for a description).

26.5.14 Path length costing

The path length costing is a matching-based costing. The cost is inversely proportional to the number of particles in the path created by matching p1 to p2.

26.5.15 Polynomial velocity costing

The polynomial velocity costing is a matching-based costing. A particle path is constructed using p1, p2 and all of the particles linked to p1 in previous frames, and p2 in subsequent frames. Polynomials are fitted, in a least squares sense, to the *x* and *y* coordinates of particles in the path, as functions of the frame number. The cost is proportional to the error in this fit. The error can be based on the difference between the actual particle positions and those predicted by the polynomial fit for all particles in the path, or for just particles p1 and p2.

The user must set the following parameters in the details view:

- *Maximum future/past matches used* – the maximum number of particles in subsequent or past frames that can be used to evaluate the least-squares polynomial. The default, and maximum, value is *5*, the minimum value is *1*.

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 26.5.11 for a description).
- *Local error* – is this radio button is pushed the error is based only on the locations of p1 and p2.
- *Total error* – is this radio button is pushed the error is based only on the locations of all particles in the path.

26.5.16 Recent acceleration costing

The recent acceleration costing is a matching-based costing. Its cost is based on the difference between the acceleration of p1 deduced from its match to p2, and an estimated acceleration based on previous matches of p1 or subsequent matches to p2 (or both if they are available). It is very similar to the recent velocity costing but works with acceleration instead.

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 26.5.11 for a description).

26.5.17 Recent velocity costing

The recent velocity costing is a matching-based costing. Like the local velocity costing it uses an estimate of the velocity of p1 to predict its position in frame 2 and bases the cost on the distance between p2 and the predicted location of p1. Where the recent velocity costing differs is in its calculation of the velocity estimate. It uses p1's match in the previous frame, or p2's match in the next frame (or both if they are available) to estimate the velocity.

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm

length is used to normalise the cost (see section 26.5.11 for a description).

26.5.18 Space average acceleration costing

The space average acceleration costing is a matching-based costing that is similar to the local velocity costing. The acceleration of particle p1, based on its match to p2, is compared to an estimated acceleration, based on an average of the particle accelerations in a window centred on p1.

The user must set the following parameters in the details view:

- *Window width (mm)* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height (mm)* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.
- *Use space average velocity* – if this check box is ticked then a space average velocity costing (local velocity costing) is used as the default if an acceleration cannot be calculated. This might arise if insufficient matches are available to calculate an acceleration.

26.5.19 Time average velocity costing

The time average velocity costing is a matching-based costing. It calculates the cost in the same way as the local and recent velocity costings except it differs in the way the velocity of p1 is predicted. The time average velocity costing uses a weighted average of velocities of particles linked to p1 in previous frames, or linked to p2 in subsequent frames. Thus an average over time is found.

The relative weight allocated to the velocity estimates of previous, or subsequent, particles in the path, in calculating the average velocity over the path, is $1/i$, where i is the number of matches between the particle and p1 or p2.

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 26.5.11 for a description).

27 PTV 3D Analysis Pipelines

27.1 Description

A PTV 3D analysis pipeline is a bound process that supports the particle matching optimization system for particle 3D records described in the *System Theory and Design* manual. The pipeline comprises a number of PTV 3D analysis objects that implement the matching process. Each analysis employs one or more costing strategies for determining the costs to be used in the optimization process. The PTV 3D analyses are executed sequentially so that the particle matches produced by the first analysis will be used as the basis for the matches produced by the second and so on.

When a PTV 3D analysis pipeline is first created it is given a default name and appears in the process view of the entity to which it is attached. Changes to the pipeline, including the creation of analysis objects and their associated parameters, are achieved through the pipeline's various views, accessed via its popup menu.

27.2 Views

The PTV 3D analysis pipeline object has two views.

- *Details view.*
- *Analysis view.*

27.2.1 Details view

Description

The details view displays the name of the process and enables the user to change it.

Access

The PTV 3D analysis pipeline details view can be accessed through the pipeline's popup menu.

27.2.2 Analysis view

Description

The analysis view gives the user control over the PTV 3D analyses that are included in the pipeline. It is the default view. The analyses currently defined, and their statuses (WAITING, ABORTED or EXECUTED), are displayed in the list box at the top of the view. The status of a PTV analysis simply indicates to the user whether the current analysis has been executed. This status does not restrict the user's actions. For example a completed analysis can be executed again without resetting it first. Analyses can be edited after they have executed and so on. The buttons at the bottom of the view provide the following functions:

- *New* – creates a new PTV 3D analysis object and adds it to the list.
- *Load* – loads a previously saved PTV 3D analysis object from disk and adds it to the list. These have a file extension of .ptv3.
- *Delete* – deletes a selected PTV 3D analysis object from the list. The user is prompted to confirm this action.
- *Reset* – the *Reset* button resets all analyses to a WAITING status.
- *Clear* – this button clears all analyses from the list. The user is prompted to confirm this action.
- *Delete matches* – if the pipeline is associated with a particle record this button deletes all matches currently defined for the record. The user is prompted to confirm this action. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.
- *Execute* – if the pipeline is associated with a particle record this button executes all analyses sequentially. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the PTV 3D analysis pipeline, or by double clicking the object.

27.3 Popup menu

The PTV analysis pipeline provides the following options in its popup menu.

- *Save as…* – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the pipeline.

- *Change name…* – opens a dialog box that enables the user to change the name of the pipeline.
- *Open analysis view* – opens the object's analysis view.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the pipelines currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

27.4 PTV 3D analysis

A PTV 3D analysis object can be created in the PTV 3D analysis pipeline analysis view by pressing the New button. The new analysis is added to the pipeline and the user changes its settings, including creating costings, through its details view.

27.4.1 Details view

The details view includes four tabs, and are discussed in the following sections.

Analysis

The analysis tab allows the user to specify a name for the analysis being defined. This name is generated automatically by *Streams* whenever the user changes the selected costings. The automatic name includes the selected costings, and the process type. The user should only change this name once the process type and costings have been chosen.

Four types of process can be performed by a PTV 3D analysis object. The first is a global optimisation process by which particles in one frame are matched to those in the next, or previous, frame, in an optimal way such that the total cost of the particle matches is minimised. This is the default process. The second process is referred to as a local optimisation. In this case matches are made between particles only when the match is the optimal match for the particles in both frames. In other words, when all possible matches for a particle in frame 1 are considered the only possible match is to the particle in frame 2 that carries the lowest cost. That match will only be made if no other particle in frame 1 can be matched to the frame 2 particle with a lower cost. This process will generally lead to fewer matches than the global optimisation but it should, provided appropriate costings have been

chosen, generate better quality matches. The third process is a residual optimization in which only the currently unmatched particles participate. In this process the current matches are left unaffected. Finally it is also possible for the PTV analysis to perform a cleanup process. A cleanup uses the defined costing strategies to break matches that are currently defined if their cost exceeds the MMC. No new matches are created. The cleanup is particularly useful in removing incorrect matches made by a previous analysis. A number of the parameters defined for the PTV 3D analysis object are irrelevant when a cleanup is performed - for example the search window (although the distance costing does use this parameter).

Costings

The costings tab allows the user to choose the 3D costing strategies that the PTV object will employ in calculating particle matching costs. The user can select any number of costing strategies. Each type of costing has a number of parameters (see below) that are specified by the user in its details view. This tab has the following elements:

- *Costings defined for this analysis* – this list box lists all of the costings currently defined for the analysis. Costings can be accessed through their popup menus. This list box supports drag and drop for costings.
- *New* – presents a dialog box from which the user can select a new costing. The costing is added to the list box (see above) and the user can change the costing parameters through its details view.
- *Load* – loads a previous saved costing from disk. Costings have a *.ctg* extension.
- *Delete* – deletes all costings currently selected in the list. The user is asked to confirm this action.
- *Clear* – clears all costings from the list. The user is asked to confirm this action.

Optimisation

The majority of the parameters governing the particle matching process are set in the optimisation tab of the PTV 3D analysis details view.

The following parameters are set here:

- *MMC* – this is the maximum matching cost. If the cost of matching a particle in frame 1 to a particle in frame 2 exceeds this value the match is not considered in the optimisation process.
- *Search window* relative to particle position – the search window defines a 3D rectangular window, positioned relative to a particle in frame 1. Only particles lying within this window in frame 2 are considered as possible candidate matches for the frame 1 particle. This window allows the user

to severely restrict the number of particles that are considered as possible matches to any particle in frame 1, thus vastly reducing the processing time required to perform the optimisation. The window coordinates are in spatial unit of the record. Note the window is defined by the position of its bottom, left, back corner relative to p1 (negative values mean to the left of, below and behind, p1), and its width, height, and depth and its definition is independent of the direction of iteration through the frames. By default the search window is centred on the particle and has a width, height and depth that are 10% of the record's window width, height and depth. It is also possible for the user to choose to exclude a spherical region, centred on the particle, from the search window. The radius of the excluded region is specified in the Exclusion radius parameter. This feature can be useful in eliminating stationary particles from the matching process.

- *Ensure forward and backward matches agree* – by checking this box the user restricts the matches only to those that agree if the frames are analysed in both directions – is matching particles in frame 1 to those in frame 2 and then matches those in frame 2 to those in frame 1. For many PTV analyses these two matching processes will yield the same results (e.g. with a search window centred on the particle and using a state based costing such as distance costing) and hence checking this box will have no effect.
- *Number of iterations* – while the problem of matching particles in one frame to those in the next is a linear programming problem, the problem of matching particles throughout all frames is not. This is due to the fact that some of the costing strategies use matches from previous, or subsequent frames, in the calculation of their costs. Therefore changes to the matches in one frame can affect the desirability of matches in another. The optimisation process, that works through the frames in the record in sequence making matches according to the chosen costings etc, can iterate through the frames as many times as the user wishes. This parameter specifies the number of iterations.
- *Alternating forward/backward iterations* – if more than one iteration through the frames is performed the user can select whether each iteration is in the same direction through the frames – eg starting at frame *0* and finishing at frame *N-1* every time – or whether the direction should alternate.
- *Initial direction* – the user can choose whether the optimisation process starts at frame *0* and progresses forward through the frames, or starts at frame *N-1* and progresses backward through the frames.
- *Maximum number of sweeps* – it is possible that the auction algorithm does not converge for a particular set of strategies when processing a certain frame. In this case the maximum number of sweeps through the auction algorithm (see *System Theory and Design* Manual) is specified here.
- *All frames* – when the pipeline is associated with an entity other than a particle record the number of frames in the record will not be known ahead of time. This check box tells the system to process all of the available frames.
- *First frame / last frame* – the PTV analysis can be restricted to a subset of the frames in the record. Frames outside this range are not considered in the analysis.

Regions

On occasions it can be desirable to limit the particles that are analysed within a frame. The regions tab enables the user to limit the particles involved in the matching process to user specified regions. Regions for both the first and second frame in the matching process can be set. Only particles that lie within these regions will participate in the matching process.

The controls in this tab are

- *First frame regions* – this list box displays the regions used to limit the particles in the first frame involved in the matching process. If this list box is empty all particles participate. If multiple regions are listed then particles lying within any of them will be matched if possible. This list box supports drag and drop of 3D regions.
- *Load* – this button allows the user to load regions from a disk file into the *first frames regions* list box. These regions are typically created within the particle 3D record particle view (see section 9.2.2).
- *Delete* – all selected regions in the *first frame regions* list box are deleted when this button is pressed.
- *Second frame regions* – this list box displays the regions used to limit the particles in the second frame involved in the matching process. If this list box is empty all particles participate. If multiple regions are listed then particles lying within any of them will be matched if possible. This list box supports drag and drop of 3D regions.
- *Load* – this button allows the user to load regions from a disk file into the *second frames regions* list box. These regions are typically created within particle 3D record particle view (see section 9.2.2).
- *Delete* – all selected regions in the *second frame regions* list box are deleted when this button is pressed.

27.4.2 Popup menu

The PTV 3D analysis object provides the following options in its popup menu.

- *Save as…* – saves the analysis object in a disk file. All analysis object files have the extension *.ptv3*.
- *Set status* – changes the status of the analysis object.
- *Open details view* – opens the details view so that the user can change the parametes described in the tabs above.

27.5 Costings in 3D

This section summarises the various costing strategies that are available when a PTV 3D analysis object attempts to match particles. The costings are very similar to their 2D equivalents and they are discussed in the *System Theory and Design* Manual. We will not repeat these discussions here but would refer the reader to that document.

In the sections that follow, the following terminology will be used. Matches are being sought between two frames, frame 1 and frame 2. P1 is an arbitrary particle in the first frame of the pair being analysed, and a match with an arbitrary particle in the second frame, p2, is being considered.

All costings have a single details view. They do not have a popup menu.

27.5.1 Details View

Description

The details view displays the costing's parameters. All costing details views have two tabs, one which is common to all and the other which is particular to each costing. The common tab includes the following:

Name

- *Name* – the user specified name of the costing. The default is the costing type.
- *Weighting* – the cost is multiplied by this weighting to obtain the total contribution this costing makes to the total cost in a PTV 3D analysis. This parameter is particularly useful when more than one costing is defined in a PTV 3D analysis.
- *Use as default only* – if ticked this costing won't be used unless no other costing is available.

For each costing described below the parameters, peculiar to that costing will be described.

Access

A costing details view can be accessed through the settings button on the Costing tab of the PTV 3D analysis details view.

27.5.2 Adjacency costing

The adjacency costing is a state-based costing that relies on matching particle patterns between frames. Particles that lie within a 3D rectangular region centred on p1 are displaced the same amount as p1 as it moves to p2, and compared with the particles lying within the same 3D rectangular region centred on p2. The distances between the displaced frame 1 particles and their closest neighbours (adjacent particles) in frame 2 comprise the cost.

Parameters

- *Window width* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Window depth* – the depth of the window that defines which particles are used in the cost calculation in each frame. By default the window depth is 10% of the record window's depth.
- *Max distance* – if the closest particle in the next frame is further away than this distance then the distance does not contribute to the cost.
- *Minimum norm* – normally the distance between p1 and p2 is used to normalize the cost. If the user chooses to limit how small this normalizing factor can be then this value is used instead.
- *Default cost* factor – if this cost can't be calculated, because there are no adjacent particles, the cost is defined to be this value.
- *Use minimum norm* – normally the distance between p1 and p2 is used to normalize the cost. If this checkbox is selected the minimum norm is used.
- *Only one match* – sometimes two particles from frame 1 will have the same adjacent particle in frame 2. If this check box is ticked only one of the frame 1 particles can have the frame 2 particle as its adjacent particle. Otherwise it can be adjacent to both.
- *Use max distance as default* – if a frame 1 particle has no adjacent particle in frame 2 then the costing can treat this particle in two ways. If this checkbox is ticked then the maximum distance is used as the default distance to the adjacent particle and this distance is included in the costing. If this checkbox is not selected then the particle is ignored and does not contribute to the costing.
- *Use weights* – if this check box is ticked the contributions of each frame 1 particle to the total cost will be weighted by the inverse of its distance from p1. Thus closer particles carry more weight in the determination of the total cost.

27.5.3 Centre of mass costing

The centre of mass costing is a state-based costing that uses the distribution of particles in each frame as a measure of the cost. The location of the centre of mass of the particles that lie within a 3D rectangular region centred on p1 is compared with the position of the centre of mass of particles within the same rectangular region centred on p2. The size of the particles is not considered in the centre of mass calculation. The cost is the distance between the centre of mass in frame 1, incremented by the displacement between p1 and p2, and the centre of mass in frame2. It is normalised by the distance between p1 and p2.

Parameters

- *Window width* – the width of the window that defines which particles are used in the centre of mass calculation in each frame. By default the window width is 10% of the clip window's width.
- *Window height* – the height of the window that defines which particles are used in the centre of mass calculation in each frame. By default the window height is 10% of the clip window's height.
- *Window depth* – the depth of the window that defines which particles are used in the cost calculation in each frame. By default the window depth is 10% of the record window's depth.
- *Default cost factor* – if this cost can't be calculated because there are no particles within the window in frame 1 or frame 2 then this cost value is used as the default.

27.5.4 Colour costing

The colour costing is a state-based costing that uses the difference in particle colour as a measure of the cost. Each colour component can be weighted so that it contributes more or less to the cost (see the *System Theory and Design* manual).

Parameters

- *Red weight* – the absolute difference between the red intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.
- *Green weight* – the absolute difference between the green intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.
- *Blue weight* – the absolute difference between the blue intensity of the two particles is multiplied by this factor before the RMS of the colour

differences is computed and normalised. The default value is 1 and the range is 0-1.

27.5.5 Distance costing

The distance costing is a state-based costing, depending on the distance of p2 from a predicted position of p1. By default the predicted position of p1 is simply its original position in frame 1, but the distance costing now allows the user to describe the velocity field in which the particle is moving with mathematical formulae, thus providing a rough estimate of its likely new position. The user also has the choice of biasing the distance to have a heavier dependency on the discrepancy in the x displacement or the y displacement

Parameters

- *Power* – the power to which the normalized distance is raised in computing the cost. It must be positive.
- *Reference length* – the length used to normalize the distance between particles in order to compute the cost.

Formulae

- *x velocity formula* – a formula that predicts the x velocity of the flow field. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.
- *y velocity formula* – a formula that predicts the y velocity of the flow field. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.
- *z velocity formula* – a formula that predicts the z velocity of the flow field. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual.

27.5.6 Intensity costing

The intensity costing is a state-based costing. The cost is proportional to the difference in intensity of the two particles.

27.5.7 Pseudo-correlation costing

The pseudo-correlation costing has much in common with the adjacency costing but tries to mimic the 2D correlation costing in the manner in which it computes its cost. Like the adjacency costing adjacent particles

are found in the two frames. The cost in the pseudo-correlation costing is the average of costs of the adjacent particle pairs. These costs are computed as the distance between the particle centres normalized by the sum of the radii of the particles. If the distance between the centres exceeds the sum of the radii (i.e. the particles do not touch) the distance is set to the sum of the radii. Thus the maximum cost is 1 and minimum is 0, consistent with the correlation costing.

This costing also provides the possibility of increasing or shrinking the radii of the particles in the same way as the 2D correlation costing.

Parameters

- *Window width* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Window depth* – the depth of the window that defines which particles are used in the cost calculation in each frame. By default the window depth is 10% of the record window's depth.
- *Max distance* – if the closest particle in the next frame is further away than this distance then the distance does not contribute to the cost.
- *Default cost* factor – if this cost can't be calculated, because there are no adjacent particles, the cost is defined to be this value.
- *Radius factor* – the factor by which all particle radii are multiplied before the cost is calculated.
- *Only one match* – sometimes two particles from frame 1 will have the same adjacent particle in frame 2. If this check box is ticked only one of the frame 1 particles can have the frame 2 particle as its adjacent particle. Otherwise it can be adjacent to both.
- *Use weights* – if this check box is ticked the contributions of each frame 1 particle to the total cost will be weighted by the inverse of its distance from p1. Thus closer particles carry more weight in the determination of the total cost.

27.5.8 Direction velocity costing

The direction velocity costing is a matching-based costing. It uses the match to p1 in the previous frame and perhaps the match to p2 in the frame after frame 2 to compute the change in direction of the particle trajectory. Specifically it measures the change in angle of the trajectory if p1 is matched to p2. This angle is normalized by a reference angle specified by the user to compute the cost. Thus, if the particle suffers no

change in direction, independent of the change in velocity magnitude, then the cost is zero.

The user can choose whether to use both the match prior to frame 1 as well as the match subsequent to frame 2 to estimate the change in angle in the trajectory, or just use the match prior to frame 1.

The user must set the following parameters in the details view:

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no direction change estimate is available.
- *Norm angle (deg)* – this angle is used to normalize the computed angle to produce a cost.
- *Use before and after* – if this box is ticked the angle measuring the change in direction is the average of that between the displacement from the particle matched to p1 in the previous frame to p1 and the displacement from p1 to p2, and that between p1 and p2 and the displacement from p2 to the particle it is matched to in the subsequent frame. This assumes both matches exist. If not the angle will be based on either one or the other of the changes in direction. If this box is not checked only the first of these two angles is used.

27.5.9 Least squares velocity costing

The least squares velocity costing is a matching-based costing. It uses the velocity estimates at neighbouring particles to calculate a three-dimensional polynomial fit to each of the three velocity components in the region near p1. These three-dimensional polynomials can be of any order from 0 to 4 and are based on a least squares fit to the neighbouring velocities. These polynomials are used to estimate the velocity at the location of p1 and hence to predict the likely displacement of p1 between frames 1 and 2. The cost is proportional to the distance between p2 and the predicted location of p1.

While this costing is nominally a velocity based costing, actually it is particle displacements that are used in the cost calculation. The cost is normalised by the predicted displacement of p1. For regions of very low velocity this might be very small with the result that the costs are amplified in an undesirable way. This costing, like many of the velocity based costings, allows the user to nominate a default normalisation length. If the displacement of p1 is less than this value the default length is used.

Parameters

- *Polynomial order* – the order of the two-dimensional polynomials used to fit the velocity components in the region near p1.
- *Number of neighbours* – the number of neighbouring particles used in the calculation of the polynomial fit to the local velocity field. These particles are the nearest neighbours to p1.
- *Maximum distance* – any nearest neighbours that are further from p1 than this distance will be excluded from the least squares polynomial fit to the velocity field.
- *Minimum cost weight* – each velocity used in calculating the velocity polynomial fits are multiplied by a weight factor. This parameter is the weight factor used for the velocity of the particle furthest from p1. The value must lie between 0 and 1. The weight factor for other velocities decreases linearly from 1, for a particle at p1, to the minimum cost weight at the most distant particle.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length(mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.

27.5.10 Local velocity costing

The local velocity costing is a matching-based costing. It uses a local estimate of the velocity in the neighbourhood of p1 to predict the location of p1 in frame 2.The cost is proportional to the distance between p2 and the predicted location of p1.

The local estimate of the velocity for p1 can be obtained in two ways. In the first instance the user can specify a 3D rectangle, centred on p1, and the velocity is based on the current matches of all particles within this rectangle in frame 1 that are currently matched. Either a weighted average of the velocities is used (the weight varies linearly from 1 at the location of p1 to 0 at the corners of the specified rectangle) or their median value. The second of these is likely to be more accurate if some of the velocities within the rectangle are incorrect. In the second instance the user can choose to use the matches of a number of nearest neighbours to p1 in frame 1. In choosing this option the user specifies a rectangle that all of the nearest neighbor particles must lie within. Again the velocity can be computed using a median or weighted average.

The nearest neighbor option can be particularly useful if the particle density within the image is not uniform. Using a fixed rectangle that is suitable for densely seeded regions of the flow is likely to lead to no velocity estimate at all in sparely seeded regions. The nearest neighbor

option ensures that a velocity estimate will be available in almost all cases while at the same time providing some limit on the particles used in obtaining this estimate.

The local velocity costing uses a default normalization length in the same way the least squares velocity costing (see section 27.5.9).

Parameters

- *Window width* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Window depth* – the depth of the window that defines which particles are used in the cost calculation in each frame. By default the window depth is 10% of the record window's depth.
- *Use nearest neighbours* – select this check box if the nearest neighbours option is to be used.
- *Number of neighbours* – the number of nearest neighbours used to computer the velocity estimate. Only relevant when the Use nearest neighbours check box is selected. Note that fewer nearest neighbours will be used if there are insufficient particles in the specified bounding window.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.
- *Use weighted average* – if this radio button is checked then a weighted average is used to estimate the velocity.
- *Use median* – if this radio button is checked then the median is used to estimate the velocity.

27.5.11 Magnitude velocity costing

The magnitude velocity costing is a matching-based costing that uses the change in velocity magnitude as the basis for the cost. The magnitude of the velocity based on matching p1 to p2 is compared to a second velocity magnitude using either a previous match for p1 or a subsequent match for p2 (or both is they are available).

Parameters

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.

- *Default norm length* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 27.5.9 for a description).

27.5.12 Path length costing

The path length costing is a matching-based costing. The cost is inversely proportional to the number of particles in the path created by matching p1 to p2.

27.5.13 Polynomial velocity costing

The polynomial velocity costing is a matching-based costing. A particle path is constructed using p1, p2 and all of the particles linked to p1 in previous frames, and p2 in subsequent frames. Polynomials are fitted, in a least squares sense, to the *x*, *y* and z coordinates of particles in the path, as functions of the frame number. The cost is proportional to the error in this fit. The error can be based on the difference between the actual particle positions and those predicted by the polynomial fit for all particles in the path, or for just particles p1 and p2.

Parameters

- *Maximum future/past matches used* – the maximum number of particles in subsequent or past frames that can be used to evaluate the least-squares polynomial. The default, and maximum, value is *5*, the minimum value is *1*.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 27.5.9 for a description).
- *Local error* – is this radio button is pushed the error is based only on the locations of p1 and p2.
- *Total error* – is this radio button is pushed the error is based only on the locations of all particles in the path.

27.5.14 Recent acceleration costing

The recent acceleration costing is a matching-based costing. Its cost is based on the difference between the acceleration of p1 deduced from its match to p2, and an estimated acceleration based on previous matches of p1 or subsequent matches to p2 (or both if they are available). It is very similar to the recent velocity costing but works with acceleration instead.

Parameters

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 27.5.9 for a description).

27.5.15 Recent velocity costing

The recent velocity costing is a matching-based costing. Like the local velocity costing it uses an estimate of the velocity of p1 to predict its position in frame 2 and bases the cost on the distance between p2 and the predicted location of p1. Where the recent velocity costing differs is in its calculation of the velocity estimate. It uses p1's match in the previous frame, or p2's match in the next frame (or both if they are available) to estimate the velocity.

Parameters

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 27.5.8 for a description).

27.5.16 Space average acceleration costing

The space average acceleration costing is a matching-based costing that is similar to the local velocity costing. The acceleration of particle p1, based on its match to p2, is compared to an estimated acceleration, based on an average of the particle accelerations in a window centred on p1.

Parameters

- *Window width* – the width of the window that defines which particles are used in the cost calculation in each frame. By default the window width is 10% of the record window's width.
- *Window height* – the height of the window that defines which particles are used in the cost calculation in each frame. By default the window height is 10% of the record window's height.
- *Window depth* – the depth of the window that defines which particles are used in the cost calculation in each frame. By default the window depth is 10% of the record window's depth.
- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.

- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost.
- *Use space average velocity* – if this check box is ticked then a space average velocity costing (local velocity costing) is used as the default if an acceleration cannot be calculated. This might arise if insufficient matches are available to calculate an acceleration.

27.5.17 Time average velocity costing

The time average velocity costing is a matching-based costing. It calculates the cost in the same way as the local and recent velocity costings except it differs in the way the velocity of p1 is predicted. The time average velocity costing uses a weighted average of velocities of particles linked to p1 in previous frames, or linked to p2 in subsequent frames. Thus an average over time is found.

The relative weight allocated to the velocity estimates of previous, or subsequent, particles in the path, in calculating the average velocity over the path, is $1/i$, where i is the number of matches between the particle and p1 or p2.

Parameters

- *Default cost factor* – this factor multiplies the weight when the cost is calculated for a particle for which no velocity estimate is available.
- *Default norm length (mm)* – if the normalisation length in the cost calculation falls below the default norm length, then the default norm length is used to normalise the cost (see section 27.5.9 for a description).

28 Bundle PTV Analysis Pipelines

28.1 Description

A bundle PTV analysis pipeline is one of two bound processes that supports 3D particle identification. The other, preferred option, is described in the following chapter. A bundle PTV analysis pipeline uses the multiple 2D perspectives provided by a bundle 2D particle record, and its fundamental concepts are described in the *System Theory and Design* manual. It is important to realize that these pipelines perform 3D particle field construction, not tracking of particles in space in order to generate three dimensional velocity fields. The PTV nomenclature associated with these pipelines arises from the fact that the 3D particle identification process uses almost identical algorithms to those employed to do particle tracking.

The pipeline comprises a number of bundle matching analysis objects that implement the particle identification process. Like other analysis objects that arise in PTV analysis pipelines these include costings to guide the optimization process. In this case the costings are used to deduce whether two particles in two of the 2D perspectives correspond to the same particle in 3D space, and if so where that particle is located.

28.2 Views

The bundle PTV analysis pipeline object has two views.

- *Details view.*
- *Analysis view.*

28.2.1 Details view

Description

The details view is the default view for the process and has five tabs, Name, Record, Parameters, Frames and Regions.

Name

- *Process name* – the name allocated to the bundle PTV analysis pipeline.

Record

- *Particle record name* – the name of the resulting particle 3D record.

Parameters

- *Number of agreements* – the analysis pipeline analyses the perspectives in pairs. This parameters specifies how many perspectives must be in agreement regarding the location of a point in 3D space before it is accepted. The larger this number the more stringent the criteria and the fewer (but probably better) particles identified. The smaller the number the less stringent the criteria and the more particles identified. This number must be at least 2 and cannot be larger than the number of perspectives.
- *Tolerance* – this parameter (using the spatial unit of the bundle) designates the maximum distance two 3D particles can be separated before they are treated as not being the same particle and hence aren't in agreement. The smaller the tolerance the more stringent the requirement for multiple perspectives to agree on a 3D particle.

Frames

- *Use all frames* – if ticked the pipeline will analyse all of the frames in bundle. If not ticked the frames are limited to those specified in the next two parameters.
- *First frame* – the first frame of the bundle to be analysed.
- *Last frame* – the *last frame of the bundle to be analysed*.

Regions

- *Excluded regions* – a list of 3D regions. All particles located in one or more of these regions will be excluded from the resulting particle 3D record. This list box supports drag and drop of 3D regions.
- *Load* – loads one or more 3D regions from a file with an *.rgn3* extension and adds them to the list of excluded regions.
- *Create* – allows the user to create a 3D region. A dialog box is displayed that prompts the user to specify the type of region. An input dialog box is then provided.
- *Delete* – deletes all of the regions currently selected in the list.

The two buttons at the bottom of the view allow the user to save the changes made or to cancel the changes.

Access

The bundle PTV analysis pipeline details view can be accessed through the pipeline's popup menu, or by double clicking the process object.

28.2.2 Analysis view

Description

The analysis view gives the user control over the bundle matching analyses that are included in the pipeline. The analyses currently defined, and their statuses (WAITING, ABORTED or EXECUTED), are displayed in the list box at the top of the view. The status of an analysis simply indicates to the user whether the current analysis has been executed. This status does not restrict the user's actions. For example a completed analysis can be executed again without resetting it first. Analyses can be edited after they have executed and so on. The buttons at the bottom of the view provide the following functions:

- *New* – creates a new bundle analysis object and adds it to the list.
- *Load* – loads a previously saved bundle analysis object from disk and adds it to the list. These have a file extension of *.ptvbun*.
- *Delete* – deletes a selected bundle analysis object from the list. The user is prompted to confirm this action.
- *Reset* – the *Reset* button resets all analyses to a WAITING status.
- *Clear* – this button clears all analyses from the list. The user is prompted to confirm this action.
- *Execute* – if the pipeline is associated with a particle record this button executes all analyses sequentially. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the bundle analysis pipeline.

28.3 Popup menu

The bundle analysis pipeline provides the following options in its popup menu.

- *Save as…* – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the pipeline.
- *Change name…* – opens a dialog box that enables the user to change the name of the pipeline.
- *Open analysis view* – opens the object's analysis view.
- *Open details view* – opens the object's details view.

- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the pipelines currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

28.4 Bundle matching analysis

A bundle matching analysis object can be created in the bundle analysis pipeline analysis view by pressing the New button. The new analysis object is created an added to the list of analyses. The user can then modify the newly created analysis through its details view.

28.4.1 Details view

The details view includes three tabs, and these are discussed in the following sections.

Analysis

The analysis tab allows the user to specify a name for the analysis being defined. This name is generated automatically by *Streams* whenever the user changes the selected costings. The automatic name includes the selected costings, and the process type. The user should only change this name once the process type and costings have been chosen.

Four types of process can be performed by a bundle analysis object. These are the same as those of the PTV analysis objects as described in section 26.4.1.

Costings

The costings tab allows the user to choose the costing strategies that the bundle analysis object will employ in calculating particle matching costs. The user can select any number of costing strategies. Each type of costing has a number of parameters (see below) that are specified by the user through the costing's details view.

The four buttons at the bottom of the tab perform the following tasks:

- *New* – creates a new costing. The user is presented with a dialog box from which they can choose the costing type. The new costing is added to the list of currently defined costings.

- *Load* – loads a bundle costing from a file and adds it to the list. It has a *.ctg* extension. Note that costings for PTV analysis are not valid for a bundle matching analysis object.
- *Deletes* – deletes a selected costing in the list. The user is prompted to confirm the action.
- *Clear* – clears all costings from the list. The user is prompted to confirm their decision.

Optimisation

The following parameters are set here:

- *Maximum matching cost* – this has the same meaning as for PTV analysis objects. Any particle match that exceeds this cost will not be included in the optimization process.
- *Maximum number of sweeps* – it is possible that the auction algorithm does not converge for a particular set of strategies when processing a certain frame. In this case the maximum number of sweeps through the auction algorithm (see *System Theory and Design* Manual) is specified here.

28.4.2 Popup menu

The bundle analysis object provides the following options in its popup menu.

- *Save as…* – saves the analysis object in a disk file. All analysis object files have the extension *.ptvbun*.
- *Set status* – changes the status of the analysis object.
- *Open details view* – opens the details view so that the user can change the parameters described in the tabs above.

28.5 Costings

This section summarises the various costing strategies that are available when a bundle analysis object attempts to match particles between perspectives. The costings available in *Streams* have been discussed at length in the *System Theory and Design* Manual. We will not repeat these discussions.

In the sections that follow the following terminology will be used. Matches are being sought between two perspectives, perspective 1 and perspective 2. P1 is an arbitrary particle in the first perspective of the pair being analysed, and a match with an arbitrary particle in the second perspective, p2, is being considered.

All costings have a single details view and a popup menu with the following two options:

- *Save as…* – saves the costing in a disk file. All analysis object files have the extension *.ctg*.
- *Open details view* – opens the details view so that the user can change the parameters described in the tabs above.

28.5.1 Details View

Description

The details view displays the costing's parameters. All costing details views have two tabs, one which is common to all and the other which is particular to each costing. The common tab includes the following:

Name

- *Name* – the user specified name of the costing. The default is the costing type.
- *Weighting* – the cost is multiplied by this weighting to obtain the total contribution this costing makes to the total cost in a bundle matching analysis. This parameter is particularly useful when more than one costing is defined in a bundle matching analysis.
- *Use as default only* – if ticked this costing won't be used unless no other costing is available.

For each costing described below the parameters, peculiar to that costing will be described.

Access

A costing details view can be accessed through the settings button on the Costing tab of the bundle analysis details view.

28.5.2 Colour costing

The colour costing is a state-based costing that uses the difference in particle colour as a measure of the cost. Each colour component can be weighted so that it contributes more or less to the cost (see the *System Theory and Design* manual).

The parameters set in the details view for this costing are:

- *Red weight* – the absolute difference between the red intensity of the two particles is multiplied by this factor before the RMS of the colour

differences is computed and normalised. The default value is 1 and the range is 0-1.

- *Green weight* – the absolute difference between the green intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.
- *Blue weight* – the absolute difference between the blue intensity of the two particles is multiplied by this factor before the RMS of the colour differences is computed and normalised. The default value is 1 and the range is 0-1.

28.5.3 Distance costing

The distance costing is a state-based costing, depending on the minimum distance between the light rays that pass through p2 and p1. The user specifies a reference length used to normalize the distance in computing the cost.

The user must provide the following parameters in the details view:

- *Reference length* – the distance is divided by this length to compute the cost.

28.5.4 Intensity costing

The intensity costing is a state-based costing. The cost is proportional to the difference in intensity of the two particles.

There are no additional parameters.

28.5.5 Mapping error costing

The mapping error based costing is a state-based costing, where the cost is calculated to be the minimum mapping error of a 3D particle onto particles p1 and p2. The cost is a distance in 2D space. The cost is very expensive to compute as the determination of the location of the particle in 3D space that minimizes the mapping error requires a non-linear optimization for each choice of p2. In order to drastically reduce the calculation time a distance costing, which is computationally cheap, can be used to filter prospective matches.

The user specifies a reference length used to normalize the distance in computing the cost, and, if the distance costing is used as a filter, a reference length must also be specified for it.

The user must provide the following parameters in the details view:

- *Reference length* – the mapping error is divided by this length to compute the cost.
- *Use distance costing as filter* – if checked the distance costing will be used to pre-screen possible matches for computation. If the distance costing generates a cost greater than 1 then the cost is set to 1 and not computed at all using the mapping error costing.
- *Distance reference length* – the reference length used for the distance costing.

28.5.6 Regions costing

The regions costing bases its cost on whether the closest point to the two particles lies within or without specified 3D regions. If it lies within the regions the specified cost is allocated to the match. If it lies outside the cost is zero.

The parameters set in the details view for this costing are:

- *Cost within regions* – if the closest point lies within the designated regions the match is allocated this cost. Otherwise the cost is zero.
- *Regions* – the list of regions used in the costing calculation.
- *Load* – loads one or more 3D regions from a file and adds them to the list.
- *Create* – creates a new region. A dialog box id displayed from which the user can choose the type of region. Thence a dialog box id displayed to allow the user to enter the appropriate region parameters.
- *Delete* – deletes selected regions in the list. The user is prompted for confirmation.

28.5.7 Size costing

The size costing is a state-based costing. The cost can be computed from the relative difference in radius of p1 and p2. The larger of the two radii is used to normalize the difference in radius (which is always positive).

There are no additional parameters.

29 Predict and Verify Process

29.1 Description

The predict and verify process produces a 3D particle record from a bundle 2D particle record. It is the recommended process for creating 3D particle data, and is viewed as more robust than the bundle PTV analysis pipeline process described in the previous chapter.

The fundamental concepts underlying the predict and verify process have been described in detail in the *System Theory and Design* manual. Essentially the particle construction process involves two steps. In the first step, the prediction step, possible three dimensional particles are constructed from the particles in two of the bundle's perspectives. One particle from each perspective is selected and a potential 3D particle is found through an optimisation process that minimizes the error between the mapped location of the potential particle into the two perspectives and the actual locations of the two particles. Mapping errors above a user specified value cause the potential particle to be discarded. The second step, the verification step, involves using the remaining perspectives to verify whether a potential particle identified in step 1 corresponds to a particle in each perspective. Verification is confirmed if the distance between the mapped 3D particle and the particle in the perspective that is closest to its mapped location is less than a user specified error. Any potential 3D particle that is verified by sufficient perspectives to meet the number of agreements required by the user is designated as a valid 3D particle.

This process, in its raw form, is computationally very expensive due to the optimisation process that produces potential 3D particles. For this reason the user can choose to undertake a vetting step that precedes the particle production. It is possible in the vast majority of cases, using the mapping functions, to accurately estimate the location of the light path that originates at a particle in a perspective and passes through the particle domain (based on the domain defined in the camera calibration). Thus, the minimum distance between the light paths from the two perspective particles can be easily calculated and if this distance is greater than some user defined tolerance the computation of the location of the potential 3D particle is aborted.

While the use of this vetting process is optional it is highly recommended that it is always employed.

29.2 Views

The predict and verify object has a single view.

- *Details view.*

29.2.1 Details view

Description

The details view is the default view for the process and has five tabs, Name, Record, Parameters, Frames and Regions.

Name

- *Process name* – the name allocated to the predict and verify process.

Record

- *Particle record name* – the name of the resulting particle 3D record.

Parameters

- *Number of agreements* – the number of perspectives that must agree on the existence of a 3D particle. As two perspectives are involved in the prediction of the original 3D particle, the number of perspectives that must verify the particle is the number of agreements – 2. The larger the number of agreements the more robust the particle construction process. The number of agreements must be at least 2 and cannot be larger than the number of perspectives.
- *Max prediction error* – this parameter specifies the largest error allowed in the prediction step before the potential 3D particle is discarded as being unviable.
- *Use distance test* – if selected, the vetting process described in section 28.1 is employed to reduce the computational cost.
- *Max distance error* – the maximum distance allowed between the two particle light rays before the corresponding 3D particle is judged to be unviable.
- *Max verification error* – the maximum mapping error allowed in the verification step. A 3D particle is not verified by a perspective if the mapping error exceeds this value.

Frames

- *Use all frames* – if ticked the pipeline will analyse all of the frames in bundle. If not ticked the frames are limited to those specified in the next two parameters.
- *First frame* – the first frame of the bundle to be analysed.
- *Last frame* – the *last frame of the bundle to be analysed*.

Regions

- *Excluded regions* – a list of 3D regions. All particles located in one or more of these regions will be excluded from the resulting particle 3D record. This list box supports drag and drop of 3D regions.
- *Load* – loads one or more 3D regions from a file with an *.rgn3* extension and adds them to the list of excluded regions.
- *Create* – allows the user to create a 3D region. A dialog box is displayed that prompts the user to specify the type of region. An input dialog box is then provided.
- *Delete* – deletes all of the regions currently selected in the list.

The two buttons at the bottom of the view allow the user to save the changes made or to cancel the changes.

Access

The predict and verify process details view can be accessed through the process's popup menu, or by double clicking the process object.

29.3 Popup menu

The predict and verify process provides the following options in its popup menu.

- *Save as…* – saves the pipeline in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the process's currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

30 Create Intensity Field Process

30.1 Description

Intensity field time series objects are created by a create intensity field process. This process takes pixel intensity information from the images in an image sequence and produces a scalar field of intensity values on a three dimensional rectangular space-time grid.

The grid can be as fine as the pixel resolution of the original images, thus providing intensity information at every pixel, or it can be coarse, thus resolving only the intensities at a few pixels (those that correspond to the nodes in the grid). The resolution of the grid will be highly dependent on the information required by the user.

This process will also generate intensity spatial derivative information on the grid. The user can choose whether to base these derivatives on the intensities at the grid nodes, using a second order finite difference approximation (where possible, near the edges and sometimes in other locations only a first order scheme is available), or on adjacent pixels, again employing a second order finite difference scheme where possible. The use of the second of these is highly reliant of extremely accurate pixel intensities, and should be used with caution.

The user also has a number of other options that control the generation of the intensity field. Firstly, the user can select the images within the sequence that are to be analysed. This does not need to include all of the images – any contiguous subset is allowed. Secondly, the user can define regions that are to be excluded from the analysis. These regions generally correspond to domains within the image where spurious intensity information resides (for example a solid boundary from which light reflections occur). Thirdly, the user can choose to use an average of the pixel intensities surrounding a grid point to compute the intensity at the grid point. This can be helpful in reducing noise in the recorded images.

Finally the user must specify the intensity in which they are interested. For an RGB image they may choose to calculate the red intensity, green intensity, blue intensity or an average of all three.

30.2 Views

The create intensity field process has a single view – the details view.

30.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the interpolation scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described below.

Process

- *Process name* – the name of the create intensity field process.

Name

- *Intensity field name* – the name of the intensity field time series that is created by the process.

Fields

- *Calculate average intensity* – for an RGB image this calculates the average intensity of the pixel at the grid point.
- *Calculate red intensity* – for an RGB image this calculates the red intensity of the pixel at the grid point.
- *Calculate green intensity* – for an RGB image this calculates the green intensity of the pixel at the grid point.
- *Calculate blue intensity* – for an RGB image this calculates the blue intensity of the pixel at the grid point.
- *Calculate gray scale 10 bit intensity* – for a 10 bit gray scale image this calculates the intensity of the pixel at the grid point.
- *Calculate gray scale 12 bit intensity* – for a 12 bit gray scale image this calculates the intensity of the pixel at the grid point.
- *Calculate real value* – for a real value image this calculates the real value of the pixel at the grid point.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point. The units are those specified in the image sequence.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.

- *Interpolation pts x spacing* – the spacing between grid points in the *x* direction
- *Interpolation pts y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd2d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd2d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Intensity

- *Use binning* – when checked the intensity at a grid point will be computed by averaging the intensities at surrounding pixels based on the left, right, bottom and top bins specified below.
- *Left bin (pix)* – the *number of pixels to the left of the grid point used in computing the intensity at the grid point.*
- *Right bin (pix)* – the *number of pixels to the right of the grid point used in computing the intensity at the grid point.*
- *Bottom bin (pix)* – the *number of pixels below the grid point used in computing the intensity at the grid point.*
- *Top bin (pix)* – the *number of pixels above the grid point used in computing the intensity at the grid point.*

Derivatives

- *Grid based* – if this radio button is checked the intensity derivative is calculated using finite differences on the grid intensity values.
- *Pixel based* – if this radio button is checked the intensity derivative is calculated using finite differences on the adjacent pixel intensity values.

Frames

- *All frames* – if selected all frames in the image sequence are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frames box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the intensity field creation process. This might include regions of reflected light. These regions are most easily created in the image sequence image view. This list supports drag and drop so regions from a region manager view (see chapter 47) can be dropped here.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through create intensity field process popup menu or by double clicking on the process.

30.3 Popup menu

The create intensity field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not an image sequence he execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

31 Create Velocity 2D Field Process

31.1 Description

Velocity 2D field time series objects are created by a create velocity 2D field process. This process interpolates the particle-based velocities derived from a particle 2D record onto a three dimensional rectangular space-time grid

The user has considerable choice in how the interpolation process is to proceed. Ideally, if the particle matches are of high quality, and the particle density is high and uniform, the resulting velocity information should be insensitive to the interpolation choices. However, in most practical situations, the interpolation choices need to be made carefully and perhaps checked by comparing the results with those from other interpolation schemes.

The interpolation process has a number of key components. The first is the specification of the rectangular grid on which the particle velocities are to be interpolated. As discussed in the *System Theory and Design* manual it is important that the grid resolution is sensibly chosen. Certainly deciding on a grid that has significantly more nodes than a frame has particles would suggest that the expectations of the user may be unrealistic. The regions near the edges of a particle frame frequently contain the most error prone results due to the lack of matching information available. It is good practice to select grids that exclude these problem areas and that avoid the need for extrapolation outside the domain covered by the particles.

The second component is the interpolation scheme. As described in the *System Theory and Design* manual a number of options are available for interpolating the velocities and their derivatives. Each has its strengths and weaknesses and some exploration of the results using the different schemes may be worthwhile. The reader is referred to this manual for full details of the various interpolation schemes.

The third component is the choice of data within the particle record that is used to produce the velocity field. The grid specification already provides one mechanism for excluding data near the edges of a particle frame. A number of other mechanisms are also available. The first

involves deciding which particle velocities should be used in the interpolation. Particle velocity estimates can be made provided the particle has one match to a particle in a previous or subsequent frame. However, particle matches that are part of a particle path that includes more than two particles, are likely to be considerably more reliable. The user has the choice as to whether to insist that only particles with matches in both previous and subsequent frames are used in the velocity field calculation. A second choice allows the user to limit the particle record frames used in the analysis. And the third option enables the user to define regions within the flow where velocities will not be calculated. These regions usually correspond to internal solid boundaries where fluid velocities simply are not defined.

31.2 Views

The create velocity field process has a single view – the details view.

31.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the interpolations scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create velocity 2D field process.

Name

- *Velocity field name* – the name of the velocity 2D field time series that is created by the process.

Fields

- *Calculate velocity* – this calculates the velocities based on the matches generated by the PTV 2D analysis process.
- *Calculate true velocity* – (only available for record with true velocities – this includes both model flows and image based flows where the user has artificially created true matches). This calculates the grid velocities based on the true velocities of each particle. The interpolation process is the same as that used for the PTV estimated velocities.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point. The units are those defined for the particle record.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd2d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.gr2dd* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Velocity

- *Single estimate* – the velocity is calculated provided at least one match to the particle exists. If the particle is unmatched it does not participate in the velocity field calculations. If two matches are available both are used to calculate the velocity.
- *Double estimate* – in this case a velocity is calculated for a particle only if matches exist in both previous and subsequent frames. If only one match exists the particle is excluded from the velocity field calculations as if it were unmatched.
- *Polynomial estimate* – this option provides a rather more sophisticated velocity estimate by calculating a polynomial fit to the particle path and differentiating this with respect to time at the location of the particle. The polynomial is a least squares best fit to the path. The order of the polynomial used is given in the System Theory and Design manual.
- *Max path half length* – the maximum half length of the particle path.

Interpolation

Select velocity interpolation scheme

- *Triangle based* – if this radio button is checked the velocity interpolation is based on a Delaunay triangulation.
- *Least squares* – if this radio button is checked the velocity interpolation is based on a two dimensional polynomial approximation to the velocity field obtained using least squares.
- *Binned* – if this radio button is checked the velocity interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if this radio button is checked the derivatives are calculated from the grid velocities using a finite difference approximation.
- *Triangle least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a two dimensional polynomial approximation to the velocity field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a two dimensional polynomial approximation to the velocity field obtained using least squares.
- *Triangle constant interpolation* – if this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation over the Delaunay triangulation.

Select extrapolation option

- *Interpolate only* – if this radio button is checked no extrapolation outside the Delaunay triangulation is allowed.
- *Allow extrapolation* – if this radio button is checked extrapolation outside the Delaunay triangulation is allowed.

Least squares settings

- *Number of neighbours* – when a two dimensional polynomial approximation to the velocity field is being used this field specifies the number of neighbouring particle-based velocity estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the velocity field is being used this box specifies the polynomial order. Orders of 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.
- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit triangle size* – if this check box is ticked interpolation based on a triangulation is size limited.
- *Triangle max x* – if triangle based interpolation is size limited then the velocity at a grid point that lies within a triangle whose horizontal extent is greater than the value in this field will be undefined.
- *Triangle max y* – if triangle based interpolation is size limited then the velocity at a grid point that lies within a triangle whose vertical extent is greater than the value in this field will be undefined.
- *Use binning* – if triangle based interpolation is size limited, and this check box is ticked, then a grid point at which no velocity estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its velocity.

Frames

- *All frames* – if selected all frames in the particle record are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the velocity field calculation.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through create velocity field process popup menu or by double clicking on the process.

31.3 Popup menu

The create velocity field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

32 Create Velocity 3D Field Process

32.1 Description

Velocity 3D field time series objects are created by a create velocity 3D field process. This process interpolates the particle based velocities derived from a particle 3D record onto a four dimensional rectangular space-time grid

As for the two-dimensional case the user has considerable control over this interpolation process. Different interpolation schemes are available for the field itself and its derivatives. The user can exclude regions from the interpolation process and compute the velocity field on a contiguous subset of the record's frames. All of these details are discussed in the *System Theory and Design* manual.

One particular choice is important for the user in computing 3D velocity fields and that is whether they wish to compute and store derivative information with the field. A major issue with three dimensional velocity fields is the associated memory demands. Take for example a relatively modest velocity field where the spatial grid is 50x40x40 and the time series includes 1000 frames. The resulting field has 80 million nodes. Each piece of field information requires 4 bytes of storage and therefore to store the three velocity field components will require almost 1 Gb of memory. If all velocity nine velocity derivatives are also included this memory requirement escalates to 4Gb.

If the user chooses not to compute and store derivative information this does not mean that quantities such as vorticity and divergence of the velocity field cannot be calculated. At the cost of some computational time, velocity fields without stored derivatives will compute derivatives using a second order (where possible) finite difference approximation based on the grid values of the velocity field.

32.2 Views

The create velocity field process has a single view – the details view.

32.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the interpolations scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create velocity 3D field process.

Name

- *Velocity field name* – the name of the velocity 3D field time series that is created by the process.

Fields

- *Calculate velocity* – this calculates the velocities based on the matches generated by the PTV 3D analysis process.
- *Calculate true velocity* – (only available for record with true velocities – this includes both model flows and image based flows where the user has artificially created true matches). This calculates the grid velocities based on the true velocities of each particle. The interpolation process is the same as that used for the PTV estimated velocities.
- *Store derivatives* – when checked velocity derivatives are computed using the specified interpolation algorithm and stored with the velocity field. This will increase the memory requirements for the resulting velocity 3D field time series by a factor of 3. By default the box is not checked. Even if derivatives are not stored the calculators can still compute them using from the field values on the grid.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point. The units are those defined for the particle record.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Origin z coord* – the *z* coordinate of the back most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Grid z spacing* – the spacing between grid points in the z direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x*

increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.

- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Number of z points* – the total number of grid points in the z direction. This value is inclusive of both end grid points. Therefore the front most grid point has a z coordinate of $dz*(N-1) + z_0$, where *dz* is the z increment, *N* is the total number of grid points in the *z* direction and z_0 is the origin *z* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd3d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd3d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y and z directions. It covers the entire frame.

Velocity

- *Single estimate* – the velocity is calculated provided at least one match to the particle exists. If the particle is unmatched it does not participate in the velocity field calculations. If two matches are available both are used to calculate the velocity.
- *Double estimate* – in this case a velocity is calculated for a particle only if matches exist in both previous and subsequent frames. If only one match exists the particle is excluded from the velocity field calculations as if it were unmatched.
- *Polynomial estimate* – this option provides a rather more sophisticated velocity estimate by calculating a polynomial fit to the particle path and differentiating this with respect to time at the location of the particle. The polynomial is a least squares best fit to the path. The order of the polynomial used is given in the System Theory and Design manual.
- *Max path half length* – the maximum half length of the particle path.

Interpolation

Select velocity interpolation scheme

- *Tetrahedron based* – if this radio button is checked the velocity interpolation is based on a Delaunay 3D triangulation.
- *Least squares* – if this radio button is checked the velocity interpolation is based on a three dimensional polynomial approximation to the velocity field at the grid points obtained using least squares.
- *Binned* – if this radio button is checked the velocity interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if the store derivatives box is ticked and this radio button is checked the derivatives are calculated from the grid velocities using a finite difference approximation.
- *Tetrahedron least squares* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a three dimensional polynomial approximation to the velocity field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a three dimensional polynomial approximation to the velocity field obtained using least squares.
- *Tetrahedron constant interpolation* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation of the field over the Delaunay 3D triangulation.

Select extrapolation option

- *Interpolate only* – if this radio button is checked no extrapolation outside the Delaunay triangulation is allowed.
- *Allow extrapolation* – if this radio button is checked extrapolation outside the Delaunay triangulation is allowed.

Least squares settings

- *Number of neighbours* – when a three dimensional polynomial approximation to the velocity field is being used this field specifies the number of neighbouring particle-based velocity estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the velocity field is being used this box specifies the polynomial order. Orders of 0, 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.
- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit tetrahedron size* – if this check box is ticked interpolation based on a 3D triangulation is size limited.
- *Tetrahedron max x* – if triangle based interpolation is size limited then the velocity at a grid point that lies within a tetrahedron whose horizontal extent is greater than the value in this field will be undefined.
- *Tetrahedron max y* – if triangle based interpolation is size limited then the velocity at a grid point that lies within a tetrahedron whose vertical extent is greater than the value in this field will be undefined.
- *Tetrahedron max z*– if triangle based interpolation is size limited then the velocity at a grid point that lies within a tetrahedron whose thickness (front to back) is greater than the value in this field will be undefined.
- *Use binning* – if tetrahedron based interpolation is size limited, and this check box is ticked, then a grid point at which no velocity estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its velocity.

Segmentation

- *Number of x segments* – when undertaking the interpolation process using least squares segmentation of the particles significantly increases the processing speed. This segmentation is independent of the segmentation of the particle record itself. This field specifies the number of segments in the x direction.
- *Number of y segments* – this field specifies the number of segments in the y direction.
- *Number of x segments* – this field specifies the number of segments in the z direction.

Frames

- *All frames* – if selected all frames in the particle record are processed.

- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all 3D regions that the user wishes to exclude from the velocity field calculation. This box supports drag and drop of 3D regions.
- *Load* – allows 3D regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through create velocity field process popup menu or by double clicking on the process.

32.3 Popup menu

The create velocity 3D field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

33 Create Density Field Process

33.1 Description

Density field time series objects are created by a create density field process. This process analyses the particle information in a particle 2D record and produces a scalar field that represents the density of particles on a three dimensional rectangular space-time grid.

No interpolation scheme is necessary for the calculation of density fields. Instead a binning process is used, whereby all particles within a rectangular cell centred on each grid point, are counted, and the particle density is calculated by dividing this number by the area of the cell. The cell dimensions correspond to the node spacings in the grid for all nodes except those on the edge of the frame. For these the cell is either a quarter or half of the grid cell size.

Derivatives of the density field are also calculated during this binning process. A finite difference approximation, based on the density values at neighbouring nodes on the grid, are used for this purpose. The user can decide whether these finite difference approximations should be accurate to first or second order.

The user can choose to restrict the data used in the production of the density. Three mechanisms exist. Firstly the grid chosen by the user can cover less than the full frame. Secondly the user can limit the particle frames used in the analysis. And thirdly the user can define regions within the flow where densities are not computed.

33.2 Views

The create density field process has a single view – the details view.

33.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the binning scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create density field process.

Name

- *Density field name* – the name of the density field that is created by the process.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Grid x spacing*– the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd2d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd2d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Derivatives

- *Approximation order* – the order of the finite difference scheme used to compute the density field derivatives.

Frames

- *All frames* – if selected all frames in the particle record are processed.

- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frames box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the intensity field creation process. This might include regions of reflected light. This list box supports drag and drop of 2D regions.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through create density field process popup menu or by double clicking on the process.

33.3 Popup menu

The create density field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

34 Create Shape Field Process

34.1 Description

Shape field objects are created by create shape field processes. A create shape field process analyses the particles in a particle 2D record and produces a scalar field of the chosen shape parameter based on a three-dimensional rectangular space-time grid. The shape field is obtained by interpolating the chosen particle shape parameter (effective diameter, area or aspect ratio) onto the grid. The validity of this interpolation process depends on the assumption that these particle characteristics are single valued, continuous functions of space. For many applications this assumption may be without foundation. Thus the use of these shape fields must be made with due care.

The interpolation process has a number of key components. The first is the choice of shape parameter to use to produce the shape field. The user can select the effective diameter, the area, or the aspect ratio of the particle.

The second component is the specification of the rectangular grid onto which the particle shape parameter is to be interpolated. As discussed in the *System Theory and Design* manual it is important that the grid resolution is sensibly chosen. It is also good practice to select grids that exclude the need for extrapolation outside the domain covered by the particles.

The third component is the interpolation scheme. As described in the *System Theory and Design* manual a number of options are available for interpolating the shape parameter and its derivatives. The reader is referred to this manual for full details of the various interpolation schemes.

34.2 Views

The create shape field process has a single view – the details view.

34.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the interpolation scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create shape field process.

Name

- *Shape field name* – the name of the shape field that is created by the process.

Fields

- *Calculate diameter* – specifies that the shape field will be based on the effective diameter of the particles.
- *Calculate area* – specifies that the shape field will be based on the area of the particles.
- *Calculate aspect ratio* – specifies that the shape field will be based on the aspect ratio of the particles.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume

the grid starts at the origin. If this box is ticked the first four grid fields are disabled.

- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd*2d extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd2d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Interpolation

Select field interpolation scheme

- *Triangle based* – if this radio button is checked the shape field interpolation is based on a Delaunay triangulation.
- *Least squares* – if this radio button is checked the field interpolation is based on a two dimensional polynomial approximation to the shape field obtained using least squares.
- *Binned* – if this radio button is checked the shape field interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if this radio button is checked the derivatives are calculated from the grid shape field using a finite difference approximation.
- *Triangle least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a two dimensional polynomial approximation to the shape field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a two dimensional polynomial approximation to the shape field obtained using least squares.
- *Triangle constant interpolation* – if this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation over the Delaunay triangulation.

Select extrapolation option

- *Interpolate only* – if this radio button is checked no extrapolation outside the Delaunay triangulation is allowed.
- *Allow extrapolation* – if this radio button is checked extrapolation outside the Delaunay triangulation is allowed.

Least squares settings

- *Number of neighbours* – when a two dimensional polynomial approximation to the shape field is being used this field specifies the number of neighbouring particle-based shape field estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the shape field is being used this box specifies the polynomial order. Orders of 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.
- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit triangle size* – if this check box is ticked interpolation based on a triangulation is size limited.
- *Triangle max x* – if triangle based interpolation is size limited then the shape field at a grid point that lies within a triangle whose horizontal extent is greater than the value in this field will be undefined.
- *Triangle max y* – if triangle based interpolation is size limited then the shape field at a grid point that lies within a triangle whose vertical extent is greater than the value in this field will be undefined.
- *Use binning* – if triangle based interpolation is size limited, and this check box is ticked, then a grid point at which no shape field estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its shape field.

Frames

- *All frames* – if selected all frames in the image sequence are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frames box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the shape field creation process. This might include regions of reflected light. This list box supports drag and drop of 2D regions.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through create shape field process popup menu or by double clicking on the process.

34.3 Popup menu

The create shape field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

35 Create Displacement 2D Field Process

35.1 Description

Displacement 2D field time series objects are created by one of two processes. The first, the create displacement 2D field process, acts on a particle 2D record and interpolates a displacement field onto a three dimensional rectangular space-time grid. The second, the create displacement 2D field from paths process, acts on a Lagrangian path field 2D to produce the displacement field. The two processes appear identical to the user, but there are reasons for choosing the latter over the former as will be discussed later.

The majority of the settings available to the user in undertaking this field creation are similar to those described in chapter 31 where the creation of velocity 2D field time series was discussed. However there is one important difference that should be understood. The user has the choice of whether they wish to fix the grid in space or fix the grid to the material being observed. In the first case the grid and interpolation process mimics that for the other Eulerian fields. The displacement at each grid point is obtained by interpolating displacement data from neighbouring particles. In the second case, however the grid is locked to the material. Thus, while the material may move under loading the displacements at a fixed grid point correspond to the same fixed point in the material. Each grid point is tracked through space from frame to frame, starting from the first frame, and interpolation is done using the particles neighbouring the tracked grid point. However the interpolated displacement is associated with the grid point in its original location. Thus any grid point provides the displacement (and other information such as strain) at a fixed point in the material not a fixed point in space. The difference between these two schemes is most easily understood by considering how the data for a deforming material will be presented. Take a very simple case of a material that simply moves as a rigid body with time. In the first space-based scheme the material will, over time, move through the grid. Grid points may contain material displacements in some frames but not in other frames. In the material-based scheme the material will be fixed relative to the grid. Displacements at a grid point will be the displacements of the location in the material that was at that grid point in the first frame, and so grid points that are initially within

the material will have displacements defined for all time. In most circumstances these material based displacements are of greater interest.

Why would one choose to generate the displacement field using a Lagrangian path field rather than a particle record? The answer lies in the type of information that is typically sort in a displacement field. Displacement fields are generally of interest in the area of structural or materials testing and material-based displacements are typically the most interesting to the experimentalist. In addition, it is accumulated displacement and strain in the material at any time that is the primary focus. This is quite unlike the requirements in a fluid mechanics experiment where the instantaneous velocities in the fluid at a particular time are required. The Lagrangian path field provides the interpolation process with direct access to the accumulated displacements of the material particles. Thus the interpolation process can interpolate the accumulated or cumulative displacements directly from the particle paths. On the other hand, the particle record naturally stores, and thus interpolates, incremental displacements between frames (as is required for the computation of the instantaneous velocity field). The impact of this different representation of the displacement data is that errors in the interpolated fields are significantly smaller if the displacement field is computed from the Lagrangian path field.

35.2 Views

The two processes have a single view – the details view.

35.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the field desired, the interpolation scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the process.

Name

- Displacement *field name* – the name of the displacement 2D field time series that is created by the process.

Field

- *Material based field* – this is the default setting and corresponds to the material based displacement field described above.
- *Space based field* – if selected the displacement field will be space-based.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd2d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd2d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Interpolation

Select displacement interpolation scheme

- *Triangle based* – if this radio button is checked the displacement interpolation is based on a Delaunay triangulation.

- *Least squares* – if this radio button is checked the displacement interpolation is based on a two dimensional polynomial approximation to the displacement field obtained using least squares.
- *Binned* – if this radio button is checked the displacement interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if this radio button is checked the derivatives are calculated from the grid displacements using a finite difference approximation.
- *Triangle least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a two dimensional polynomial approximation to the displacement field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a two dimensional polynomial approximation to the displacement field obtained using least squares.
- *Triangle constant interpolation* – if this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation over the Delaunay triangulation.

Select extrapolation option

- *Interpolate only* – if this radio button is checked no extrapolation outside the Delaunay triangulation is allowed.
- *Allow extrapolation* – if this radio button is checked extrapolation outside the Delaunay triangulation is allowed.

Least squares settings

- *Number of neighbours* – when a two dimensional polynomial approximation to the displacement field is being used this field specifies the number of neighbouring particle-based displacement estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the displacement field is being used this box specifies the polynomial order. Orders of 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.
- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit triangle size* – if this check box is ticked interpolation based on a triangulation is size limited.
- *Triangle max x* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a triangle whose horizontal extent is greater than the value in this field will be undefined.
- *Triangle max y* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a triangle whose vertical extent is greater than the value in this field will be undefined.
- *Use binning* – if triangle based interpolation is size limited, and this check box is ticked, then a grid point at which no displacement estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its displacement.

Frames

- *All frames* – if selected all frames in the particle record are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the displacement field calculation. This list box supports drag and drop of 2D regions.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through the process's popup menu or by double clicking on the process.

35.3 Popup menu

The create displacement field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

36 Create Displacement 3D Field Process

36.1 Description

Displacement 3D field time series objects are created by a create displacement 3D field process. This process interpolates the particle-based displacements derived from a particle 3D record onto a four dimensional rectangular space-time grid

The majority of the settings available to the user in undertaking this interpolation are similar to those described in chapter 32 when discussing the creation of velocity 3D field time series. As with the creation of displacement 2D fields the user can choose whether to fix the grid in space or fix it to the material being observed. See section 35.1 for more details.

36.2 Views

The create displacement 3D field process has a single view – the details view.

36.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the field desired, the interpolation scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create displacement 3D field process.

Name

- Displacement *field name* – the name of the displacement 3D field time series that is created by the process.

Field

- *Material based field* – this is the default setting and corresponds to the material based displacement field described above.
- *Space based field* – if selected the displacement field will be space-based.
- *Store derivatives* – if checked the derivatives of the displacement are stored in the field. If not the derivatives are not stored and derivatives are computed when required using finite difference approximations based on the displacements at each grid point. Not storing derivatives is will reduce the memory requirements for storing the field time series by roughly 75%.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point. The units are those defined for the particle record.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Origin z coord* – the *z* coordinate of the back most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Grid z spacing* – the spacing between grid points in the *z* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Number of z points* – the total number of grid points in the *z* direction. This value is inclusive of both end grid points. Therefore the front most grid point has a *z* coordinate of $dz*(N-1) + z_0$, where *dz* is the *z* increment, *N* is the total number of grid points in the *z* direction and z_0 is the origin *z* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd3d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd3d* extension.

- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y and z directions. It covers the entire frame.

Interpolation

Select displacement interpolation scheme

- *Tetrahedron based* – if this radio button is checked the displacement interpolation is based on a Delaunay 3D triangulation.
- *Least squares* – if this radio button is checked the displacement interpolation is based on a three dimensional polynomial approximation to the displacement field at the grid points obtained using least squares.
- *Binned* – if this radio button is checked the displacement interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if the store derivatives box is ticked and this radio button is checked the derivatives are calculated from the grid displacement using a finite difference approximation. This is the only option available if the field is material based.
- *Tetrahedron least squares* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a three dimensional polynomial approximation to the displacement field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a three dimensional polynomial approximation to the displacement field obtained using least squares.
- *Tetrahedron constant interpolation* – if the store derivatives box is ticked and this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation of the field over the Delaunay 3D triangulation.

Least squares settings

- *Number of neighbours* – when a three dimensional polynomial approximation to the displacement field is being used this field specifies the number of neighbouring particle-based displacement estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the displacement field is being used this box specifies the polynomial order. Orders of 0, 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.
- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit tetrahedron size* – if this check box is ticked interpolation based on a 3D triangulation is size limited.
- *Tetrahedron max x* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a tetrahedron whose horizontal extent is greater than the value in this field will be undefined.
- *Tetrahedron max y* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a tetrahedron whose vertical extent is greater than the value in this field will be undefined.
- *Tetrahedron max z*– if triangle based interpolation is size limited then the displacement at a grid point that lies within a tetrahedron whose thickness (front to back) is greater than the value in this field will be undefined.
- *Use binning* – if tetrahedron based interpolation is size limited, and this check box is ticked, then a grid point at which no displacement estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its displacement.

Segmentation

- *Number of x segments* – when undertaking the interpolation process using least squares segmentation of the particles significantly increases the processing speed. This segmentation is independent of the segmentation of the particle record itself. This field specifies the number of segments in the x direction.
- *Number of y segments* – this field specifies the number of segments in the y direction.
- *Number of x segments* – this field specifies the number of segments in the z direction.

Frames

- *All frames* – if selected all frames in the particle record are processed.

- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all 3D regions that the user wishes to exclude from the displacement field calculation. This box supports drag and drop of 3D regions.
- *Load* – allows 3D regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through a create displacement 3D field process popup menu or by double clicking on the process.

36.3 Popup menu

The create displacement 3D field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

37 Create Displacement 3D Surface Field Process

37.1 Description

Displacement 3D surface field time series objects are created by a create displacement 3D surface field process. This process interpolates the particle-based displacements derived from a particle 3D record onto a three dimensional rectangular space-time grid, where the particle data has been projected onto the x-y plane.

The majority of the settings available to the user in undertaking this interpolation are similar to those described in chapter 35 when discussing the creation of displacement 2D field time series. See section 35.1 for details.

37.2 Views

The create displacement 3D surface field process has a single view – the details view.

37.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the field desired, the interpolation scheme, the grid spacing and size, excluded regions and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create displacement 3D surface field process.

Name

- Displacement *field name* – the name of the displacement 3D surface field time series that is created by the process.

Field

- *Material based field* – this is the default setting and corresponds to the material based displacement field described above.
- *Space based field* – if selected the displacement field will be space-based.

Grid

- *Origin x coord* – the *x* coordinate of the left most grid point.
- *Origin y coord* – the *y* coordinate of the bottom most grid point.
- *Grid x spacing* – the spacing between grid points in the *x* direction
- *Grid y spacing* – the spacing between grid points in the *y* direction
- *Number of x points* – the total number of grid points in the *x* direction. This value is inclusive of both end grid points. Therefore the right most grid point has an *x* coordinate of $dx*(N-1) + x_0$, where *dx* is the *x* increment, *N* is the total number of grid points in the *x* direction and x_0 is the origin *x* coord of the grid.
- *Number of y points* – the total number of grid points in the *y* direction. This value is inclusive of both end grid points. Therefore the top most grid point has a *y* coordinate of $dy*(N-1) + y_0$, where *dy* is the *y* increment, *N* is the total number of grid points in the *y* direction and y_0 is the origin *y* coord of the grid.
- *Use default grid spacing* – when the dimensions of the images to be processed are not available at the time the process is created the user can select this option. Using the number of points in the x and y directions the system will automatically set the grid spacing and assume the grid starts at the origin. If this box is ticked the first four grid fields are disabled.
- *Load grid* – new grid settings can be loaded from a disk file. This file has a *.grd2d* extension.
- *Save grid* – the current grid settings can be saved to a disk file. This file has a *.grd2d* extension.
- *Default grid* – the default grid settings can be reinstated with this button. This grid has 51 points in the x direction and 41 in the y direction. It covers the entire frame.

Interpolation

Select displacement interpolation scheme

- *Triangle based* – if this radio button is checked the displacement interpolation is based on a Delaunay triangulation.
- *Least squares* – if this radio button is checked the displacement interpolation is based on a two dimensional polynomial approximation to the displacement field obtained using least squares.
- *Binned* – if this radio button is checked the displacement interpolation uses binning.

Select derivative interpolation scheme

- *Grid finite difference* – if this radio button is checked the derivatives are calculated from the grid displacements using a finite difference approximation. This is the only option available when the grid is material based.
- *Triangle least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the particle positions using a two dimensional polynomial approximation to the displacement field obtained using least squares which are then interpolated onto the grid points.
- *Grid least squares* – if this radio button is checked the derivatives are obtained by calculating the derivatives at the grid points using a two dimensional polynomial approximation to the displacement field obtained using least squares.
- *Triangle constant interpolation* – if this radio button is checked the derivatives are obtained using the constant derivative estimates that arise from a linear interpolation over the Delaunay triangulation.

Select extrapolation option

- *Interpolate only* – if this radio button is checked no extrapolation outside the Delaunay triangulation is allowed.
- *Allow extrapolation* – if this radio button is checked extrapolation outside the Delaunay triangulation is allowed.

Least squares settings

- *Number of neighbours* – when a two dimensional polynomial approximation to the displacement field is being used this field specifies the number of neighbouring particle-based displacement estimates that are used in least squares fit. The minimum number of neighbours required depends on the order of the polynomial approximation.
- *Approximation order* – when a two dimensional polynomial approximation to the displacement field is being used this box specifies the polynomial order. Orders of 1, 2 and 3 are allowed.

Finite difference settings

- *Approximation order* – when finite differences are being used for the derivative calculations this box specifies the order of the finite difference approximation.

Bin settings

- *x bin* – when binning is being used this field specified the width of each bin.

- *y bin* – when binning is being used this field specified the height of each bin.

Interpolation limits

- *Limit triangle size* – if this check box is ticked interpolation based on a triangulation is size limited.
- *Triangle max x* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a triangle whose horizontal extent is greater than the value in this field will be undefined.
- *Triangle max y* – if triangle based interpolation is size limited then the displacement at a grid point that lies within a triangle whose vertical extent is greater than the value in this field will be undefined.
- *Use binning* – if triangle based interpolation is size limited, and this check box is ticked, then a grid point at which no displacement estimate is available, due to the size of its containing triangle, will attempt to use binning to obtain an estimate of its displacement.

Frames

- *All frames* – if selected all frames in the particle record are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frame s box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frame s box is selected this field is disabled.

Regions

- *Excluded regions* – this list box includes all regions that the user wishes to exclude from the displacement field calculation. This list box supports drag and drop of 2D regions.
- *Load* – allows regions to be loaded from disk into the list box.
- *Delete* – allows regions in the list to be deleted.

Access

This view is opened through a create displacement 3D surface field process popup menu or by double clicking on the process.

37.3 Popup menu

The create displacement 3D surface field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

38 Create Lagrangian 2D Path Field Process

38.1 Description

Lagrangian 2D path fields provide a Lagrangian description of two-dimensional fluid motion by storing the motion of individual particles, instead of storing velocities at fixed points in space. These fields are described in detail in chapter 20.

These fields are created by a create Lagrangian 2D field process. This process controls the paths that are included in the field. The user has considerable control over this. The path length, when it occurs and where it occurs are all at the user's discretion.

38.2 Views

The create Lagrangian 2D field process has a single view – the details view.

38.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the constraints placed on the paths (length, start and end time etc.) and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create Lagrangian 2D path field process.

Name

- *Lagrangian field name* – the name of the Lagrangian 2D path field that is created by the process.

Particle

- *Include particle shape details if available* – if this box is checked the particle shape information, aspect ratio, theta and eccentricity will be included in the particle path data if it is available.
- *Single estimate* – all matched particles are included in the path. The velocity is calculated provided at least one match to the particle exists but if two matches are available both are used to calculate the velocity.
- *Double estimate* – only particles with two matches are included in the path (ie the first and last particles are excluded). The velocity calculation is based on both matches.
- *Polynomial estimate* – all matched particles are included in the path. This option provides a rather more sophisticated velocity estimate by calculating a polynomial fit to the particle path and differentiating this with respect to time at the location of the particle. The polynomial is a least squares best fit to the path. The order of the polynomial used is given in the Syste*m Theory and Design* manual.
- *Max path half length* – the maximum half length of the particle path.

Length

- *Length lower limit* – the user can specify the minimum and maximum length of a path using this field and the next. A value of -1 means that there is no limit. All limits are measured in numbers of frames.
- *Length upper limit* – see above.

Time

- *Start time lower limit* – the path must start in or after the time specified in this box. If it started before this the earlier part of the path is removed. A value of -1 indicates no limit is imposed.
- *Start time upper limit* – the path must start in or before the time specified in this box. A value of -1 indicates no limit.
- *End time lower limit* – the path must end in or after the time specified in this box. A value of -1 indicates no limit is imposed.
- *End time upper limit* – the path must end in or after the time specified in this box. If it ended after this the later part of the path is removed. A value of -1 indicates no limit is imposed.

Space

- *No space constraints* – if selected no space constraints are placed on the paths.
- *Constrain start and end points* – if selected the start and end points of the paths must lie in the regions listed in the two list boxes labeled path must start in regions and path must end in regions. If no regions are listed then no constraint is applied. Thus the user could choose to limit the starting point of all paths by choosing one or more regions for the start but

leaving the end list of regions blank. If a path extends beyond the start and end regions, but lies within them at some point, the remainder of the path is removed. However a path that passes through the start and end regions without actually having a particle located within them for some frame will not be included in the field.

- *Constrain whole path* – if selected the user requires every path to lie entirely within the specified regions throughout its history. Paths that possess this property but which have portions of their path outside the regions will have these parts removed.
- *Path must start in regions* – the list of start regions for paths. This list box supports drag and drop of 2D regions.
- *Path must end in regions* – the list of end regions for paths. This list box supports drag and drop of 2D regions.
- *Path must lie in regions* – the list of regions that all particle paths must lie within if the user chooses to constrain the whole path. This list box supports drag and drop of 2D regions.

Frames

- *All frames* – if selected all frames in the particle record are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frames box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frames box is selected this field is disabled.

Access

This view is opened through create Lagrangian 2D field process popup menu or by double clicking on the process.

38.3 Popup menu

The create Lagrangian 2D field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.

- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

39 Create Lagrangian 3D Path Field Process

39.1 Description

Lagrangian 3D path fields provide a Lagrangian description of three-dimensional fluid motion by storing the motion of individual particles, instead of storing velocities at fixed points in space. These fields are described in detail in chapter 21.

These fields are created by a create Lagrangian 3D field process. This process controls the paths that are included in the field. The user has considerable control over this. The path length, when it occurs and where it occurs are all at the user's discretion.

39.2 Views

The create Lagrangian 3D field process has a single view – the details view.

39.2.1 Details view

Description

The details view displays all of the information specified when the process was created. This includes the constraints placed on the paths (length, start and end time etc.) and the frames included in the process. The various tabs in the details view are described.

Process

- *Process name* – the name of the create Lagrangian 3D path field process.

Name

- *Lagrangian field name* – the name of the Lagrangian 3D path field that is created by the process.

Particle

- *Single estimate* – all matched particles are included in the path. The velocity is calculated provided at least one match to the particle exists but if two matches are available both are used to calculate the velocity.
- *Double estimate* – only particles with two matches are included in the path (ie the first and last particles are excluded). The velocity calculation is based on both matches.
- *Polynomial estimate* – all matched particles are included in the path. This option provides a rather more sophisticated velocity estimate by calculating a polynomial fit to the particle path and differentiating this with respect to time at the location of the particle. The polynomial is a least squares best fit to the path. The order of the polynomial used is given in the Syste*m Theory and Design* manual.
- *Max path half length* – the maximum half length of the particle path.

Length

- *Length lower limit* – the user can specify the minimum and maximum length of a path using this field and the next. A value of -1 means that there is no limit. All limits are measured in numbers of frames.
- *Length upper limit* – see above.

Time

- *Start time lower limit* – the path must start in or after the time specified in this box. If it started before this the earlier part of the path is removed. A value of -1 indicates no limit is imposed.
- *Start time upper limit* – the path must start in or before the time specified in this box. A value of -1 indicates no limit.
- *End time lower limit* – the path must end in or after the time specified in this box. A value of -1 indicates no limit is imposed.
- *End time upper limit* – the path must end in or after the time specified in this box. If it ended after this the later part of the path is removed. A value of -1 indicates no limit is imposed.

Space

- *No space constraints* – if selected no space constraints are placed on the paths.
- *Constrain start and end points* – if selected the start and end points of the paths must lie in the regions listed in the two list boxes labeled path must start in regions and path must end in regions. If no regions are listed then no constraint is applied. Thus the user could choose to limit the starting point of all paths by choosing one or more regions for the start but leaving the end list of regions blank. If a path extends beyond the start and end regions, but lies within them at some point, the remainder of the path is removed. However a path that passes through the start and end

regions without actually having a particle located within them for some frame will not be included in the field.

- *Constrain whole path* – if selected the user requires every path to lie entirely within the specified regions throughout its history. Paths that possess this property but which have portions of their path outside the regions will have these parts removed.
- *Path must start in regions* – the list of start regions for paths. This list box supports drag and drop of 3D regions.
- *Path must end in regions* – the list of end regions for paths. This list box supports drag and drop of 3D regions.
- *Path must lie in regions* – the list of regions that all particle paths must lie within if the user chooses to constrain the whole path. This list box supports drag and drop of 3D regions.

Frames

- *All frames* – if selected all frames in the particle record are processed.
- *First frame* – the user can limit the frames processed by entering numbers into this and the next box. If the all frames box is selected this field is disabled.
- *Second frame* – the user can limit the frames processed by entering numbers into this and the previous box. If the all frames box is selected this field is disabled.

Access

This view is opened through create Lagrangian 3D field process popup menu or by double clicking on the process.

39.3 Popup menu

The create Lagrangian 3D field process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view (this view has been described above)
- *Execute* – executes the process as a standalone process. If the associated entity is not a particle record the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40 Transform Image Sequence Processes

40.1 Description

The processing of images in an image sequence is generally performed by a filter pipeline However, there are a number of transformations of the image sequence data that does not conform to the concept of a filter. These transforms are defined as image sequence transforms and are accessed as separate bound processes. As they perform different types of operations they do not conform to a transformation pipeline as defined for Eulerian fields and particle records.

The result of applying a transform to an image sequence is to produce a new image sequence. This new sequence may contain only one image, or it may contain the same number of images as the original sequence. The images contained in the new sequence may have the same dimensions as the original images or they may not. It is clear that, these types of transform do not fit into the standard definition of a filter which takes each image in an image sequence and converts it into a new image of the same size.

It is important to note that none of these transforms can execute as standalone processes unless the name of the output image sequence, and the file name of the transformed images, are defined.

Currently five image sequence transforms are defined. These are described below.

40.2 Trim transform

The trim transform allows the user to trim the images in the sequence, thus producing smaller, rectangular sub-images. This is particularly useful for saving storage space, and processing time if only a portion of the captured images is of interest. The transform has a single view, the details view.

40.2.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Trim image sequence name* – the name the user wishes to give to the process.
- *Left inset (pixs)* – this parameter specifies the number of pixels on the left hand side of the image that will be trimmed. A value of 0 ensures no pixels are trimmed.
- *Right inset (pixs)* – this parameter specifies the number of pixels on the right hand side of the image that will be trimmed. A value of 0 ensures no pixels are trimmed.
- *Top inset (pixs)* – this parameter specifies the number of pixels at the top of the image that will be trimmed. A value of 0 ensures no pixels are trimmed.
- *Bottom inset (pixs)* – this parameter specifies the number of pixels at the bottom of the image that will be trimmed. A value of 0 ensures no pixels are trimmed.
- *Image sequence name* – the name the user wishes to give the image sequence created by the transform.
- *Filename* – the user can select a filename for the transformed images by clicking on the browse button. These file names have a 5 digit suffix to number the images sequentially.
- *Browse*– the user can select the file name using this button.

Access

The details view is opened through the popup menu of the process object.

40.2.2 Popup menu

The trim process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40.3 Scale transform

The scale transform also reduced the size of the images in the sequence but does so by shrinking them not trimming them. The transform has a single view, the details view.

40.3.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Scale image sequence name* – the name the user wishes to give to the process.
- *Scale factor (0-1)* – this specifies the scale factor. A factor of 1 means the images are not shrunk.
- *Image sequence name* – the name the user wishes to give the image sequence created by the transform.
- *Filename* – the user can select a filename for the transformed images by clicking on the browse button. These file names have a 5 digit suffix to number the images sequentially.
- *Browse*– the user can select the file name using this button.

Access

The details view is opened through the popup menu of the process object.

40.3.2 Popup menu

The scale process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40.4 Average transform

The average transform condenses all of the images in the image sequence into a single image that represents the average of these images. The averaging can apply to images of any data type – RGB, gray scale or real value. In addition to the averaging process the user can choose to impose a constant velocity on the images in the sequence so that each image is offset from the one before by a constant amount. In this way the averaged image will be of larger physical dimension that the original images. When this displacement occurs, pixels in overlapping images will no longer lie on top of each other. The user can choose whether to "snap" the transformed pixel onto the new image (in other words place it at the nearest pixel location) or to use a weighted average of the pixel intensities of the pixels that partially overlap each pixel in the new image.

The constant velocity is intended to represent the motion of the camera taking the images and therefore a positive x velocity will displace images to the left and so on. The transform has a single view, the details view.

40.4.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Average image sequence name* – the name the user wishes to give to the process.
- *x velocity* – the x component of the camera velocity.
- *y velocity* – the y component of the camera velocity.
- *Sub-pixel accuracy* – when this check box is selected a weighted average of intensities of pixels overlapping each pixel in the transformed image is used to calculate the new pixel intensity.
- *Image sequence name* – the name the user wishes to give the image sequence created by the transform.
- *Filename* – the user can select a filename for the transformed images by clicking on the browse button. These file names have a 5 digit suffix to number the images sequentially.
- *Browse*– the user can select the file name using this button.

Access

The details view is opened through the popup menu of the process object.

40.4.2 Popup menu

The scale process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40.5 Rotate transform

The rotate transform allows the user to rotate the images in the sequence, thus typically producing larger rectangular images with dead space. This is particularly useful if a camera has been slightly misaligned in an experimental setup or some angled boundary is ideally viewed horizontally or vertically. The transform has a single view, the details view.

40.5.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Rotate image sequence name* – the name the user wishes to give to the process.
- *angle (degs)* – the angle of rotation (anti-clockwise) specified in degrees.
- *Fill colour* – the colour used to fill the dead space created by the rotation.
- *Choose fill colour* - The fill colour can be selected by clicking this button.
- *Image sequence name* – the name the user wishes to give the image sequence created by the transform.
- *Filename* – the user can select a filename for the transformed images by clicking on the browse button. These file names have a 5 digit suffix to number the images sequentially.
- *Browse*– the user can select the file name using this button.

Access

The details view is opened through the popup menu of the process object.

40.5.2 Popup menu

The rotate process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40.6 Subtract transform

The subtract transform produces an image sequence that is one image shorter than the original. The images in the new sequence are obtained by subtracting the pixel intensities in one image from those in the next or vice versa.

The transform has a single view, the details view.

40.6.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Subtract image sequence name* – the name the user wishes to give to the process.
- *1st from 2nd* – if chosen the first of each pair of images is subtracted from the second.
- *2nd from 1st* – if chosen the second of each pair of images is subtracted from the first.
- *Image sequence name* – the name the user wishes to give the image sequence created by the transform.

- *Filename* – the user can select a filename for the transformed images by clicking on the browse button. These file names have a 5 digit suffix to number the images sequentially.
- *Browse*– the user can select the file name using this button.

Access

The details view is opened through the popup menu of the process object.

40.6.2 Popup menu

The subtract process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

40.7 Long exposure transform

The long exposure transform produces an image sequence that contains images generated by combining a user specified number of sequential images in the original sequence. It contains the same number of images as the original. Thus the first image in the output sequence combines the first *n* images from the original sequence, where *n* is the number selected by the user. Towards the end of the image sequence a full *n* images will not be available and the last remaining images are used in the combination process. The result is a sequence of images that look like those obtained from long exposure photography.

The user is able to choose a number of methods for combining the pixel intensities in the long exposure images. This can be done using the maximum intensity, the average intensity or the sum of the intensities (subject to the possible maximum allowable intensity). For an RGB image this algorithm is used for each of the colour components separately

The transform has a single view, the details view.

40.7.1 Details View

Description

The details view displays the process information, including its name, and the transform parameters.

- *Long exposure image sequence name* – the name the user wishes to give to the process.
- *Number of frames* – the number of images to be combined into the long exposure image.
- *Average* – select if the long exposure pixel intensity is the average of pixel intensities in the combined images.
- *Sum* – select if the long exposure pixel intensity is the sum of pixel intensities in the combined images. If this sum exceeds the maximum allowable intensity then the combined intensity is set to the maximum.
- *Maximum* – select if the long exposure pixel intensity is the maximum of pixel intensities in the combined images.

Access

The details view is opened through the popup menu of the process object.

40.7.2 Popup menu

The long exposure process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

41 Transform Particle 2D Record Processes

41.1 Description

Particle 2D records can be transformed through a pipeline process that contains a number of individual transforms. The user can choose whether they wish to apply the transforms to a copy of the original particle record or to apply them to the original directly. The available transforms are listed below.

- *Camera adjustment*
- *Barrel pincushion distortion*
- *Constant velocity*
- *Swap x*
- *Swap y*
- *Remove camera movement*
- *Map coordinates*
- *Transform coordinates*

41.2 Views

Each transform pipeline has two views

- *Details view*
- *Transform view*

41.2.1 Details View

Description

The details view displays the process information, including its name, and whether a copy of the record should be used. There are two tabs:

Process

- *Process name* – the name the user wishes to give to the process.

Record

- *Create copy* – if selected the pipeline is applied to a copy of the record.
- *Particle record name* – if a copy is made then the new particle record will have this name.

Access

The details view is opened through the popup menu of the process object.

41.2.2 Transform view

Description

The transform view gives the user control over the transforms that comprise the pipeline. It is the default view. It contains the following elements:

- *Transforms defined for this pipeline* – this list box contains of the transforms currently in the pipeline. They are executed in the order in which they appear in the list.
- *New* – creates a new transform object and adds it to the list. A dialog box is displayed that prompts the user for the type of transform required.
- *Open* – loads a previously saved transform object from disk and adds it to the list. These transform files have a. *ptr* or *ptr2d* extension.
- *Insert before…* – creates a new transform an inserts it before the transform selected in the list. If no transform is selected it will be added to the end of the pipeline.
- *Delete* – deletes a selected transforms object from the list. The user is prompted to confirm this action.
- *Clear* – this button clears all transforms from the list. The user is prompted to confirm this action.
- *Execute* – if the pipeline is associated with a particle record this button executes the pipeline. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the pipeline, or by double clicking the object.

41.3 Popup menu

The transform pipeline process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open transform view* – opens the object's transform view. This is the default.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

41.4 Transforms

In this section we describe the individual transforms that are available for transforming a particle record object.

41.4.1 Views

Each transform, with the exception of the particle filter pipeline, has a single view, the details view and the details that can be set in each of these views are given in subsequent sections.

41.4.2 Popup menus

The popup menu for each field transform has the following options:

- *Saves as…* – saves the transform to a disk file with a *.ptr2d* extension.
- *Open details view* – opens the details view.

41.4.3 Adjust camera

This transform enables the particle 2D record data to be adjusted as if the original camera that captured the images had been rotated or translated.

It is often useful to adjust for minor misalignment of the camera. The particle record dimensions are unchanged during this process. So a translation or rotation may lead to particles leaving the flow domain and these particles will be removed from the record. They cannot be retrieved by an inverse transform.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x centre of rotation* – the x coordinate of the point about which the camera rotation takes place.
- *y centre of rotation* – the y coordinate of the point about which the camera rotation takes place.
- *Rotation angle (deg)* – the angle of rotation in degrees. This is the angle of the camera rotation so the particles appear to rotate in the opposite direction.
- *x translation* – the camera is translated this distance in the x direction after any rotation has taken place. The particles move in the opposite direction.
- *y translation* – the camera is translated this distance in the y direction after any rotation has taken place. The particles move in the opposite direction.

41.4.4 Barrel pincushion distortion

This transform attempts to undo lens distortion from the particle record data. The transform has the same effect as a barrel pincushion filter applied to an image filter. See section 23.4.6.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x centre* – the x coordinate of the origin used in the transformation.
- *y centre* – the y coordinate of the origin used in the transformation
- *Coeff 0* – c0 in equation 23.2. This coefficient provides a general scaling of the image without affecting the distortion.
- *Coeff 1* – c1 in equation 23.2.
- *Coeff 2* – c2 in equation 23.2
- *Coeff 3* – c3 in equation 23.2

41.4.5 Constant velocity

This transform applies a constant velocity transformation to the frames within the particle record. Thus the coordinate origin of each frame is translated by a displacement given by the product of the specified velocity and the time since the start of the record. The velocity can be thought of as that of the camera recording the particles in the record. Thus if a flow is uniform moving from left to right then a positive velocity will freeze the particles in the record.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x velocity* – the x component of the applied velocity.
- *y velocity* – the y component of the applied velocity.

41.4.6 Swap x

This transform reflects the record in a vertical line through the middle of the record. The x coordinates are swapped in each frame.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

41.4.7 Swap y

This transform reflects the record in a horizontal line through the middle of the record. The y coordinates are swapped in each frame.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

41.4.8 Remove camera movement

This transform assumes that the camera used to capture the images in the original image sequence had a superimposed, unwanted motion perhaps due to vibration. It is also assumed that this camera motion has been recorded by all particles in the frame and thus by tracking any particle,

stationary in the laboratory frame, the unwanted camera motion can be deduced. This camera motion must be stored in a vector 2D field 1D object (if both x and y movement is to be removed) or a scalar 1D field object (if only x or y movement is to be removed). The transform will effectively subtract the recorded motion from all particles. Generating the field object is most easily done using one of the calculators that is available for a Lagrangian 2D Path Field object. Ideally the field object should have the same number "frames" as the particle record being transformed.

Note that the movement information required by this transform is incremental. In other words, the movements stored in the scalar or vector field correspond to the movement from the previous frame to current frame. Thus to determine the adjustment to the particle location required for frame N, requires all previous movements to be summed.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Transform allows for* – three radio buttons are provided that allow the user to choose whether x and y movement is to be removed or only x or y movement. In the first case the vector 2D field 1D object must be provided for the transform while for the latter two a scalar 1D field is required.
- *Field file name* – the user can select the field file (vector 2D field 1D or scalar 1D field) using the Browse button.

41.4.9 Map coordinates

This transform performs a general transformation of the particle coordinates to another coordinate system. The barrel pincushion distortion transform can be viewed as a special case of this more general transformation. A particular use of this transformation in our laboratory has been to handle the case where the camera viewing direction is not normal to the particle plane. Provided suitable reference points are captured by the camera a mapping can be generated, and tablets provide specific functionality to compute this mapping.

The mapping is specified by two formulae that transform the original x and y coordinates to the mapped coordinated x' and y'.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x origin for mapping* – when the formulae for the mapping refer to x and y these coordinates are computed by subtracting the coordinates of the origin from the particle coordinates. The text box specifies the x origin. The default origin is (0,0).
- *y origin for mapping* – when the formulae for the mapping refer to x and y these coordinates are computed by subtracting the coordinates of the origin from the particle coordinates. The text box specifies the y origin. The default origin is (0,0).
- *Mapped x formula* – the formula that converts the original particle coordinates to the transformed x coordinate. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual. Note that this drag and drop mechanism is only available in the details view of the transform once the transform has been created. It is not available in the modal dialog box used to initially create the transform.
- *Mapped y formula* – the formula that converts the original particle coordinates to the transformed y coordinate. A scalar field custom function created in a tablet can be dragged and dropped into this formula field – see the User's Guide – Tablet Objects manual. Note that this drag and drop mechanism is only available in the details view of the transform once the transform has been created. It is not available in the modal dialog box used to initially create the transform.

41.4.10 Transform coordinates

This transform is intended as an alternative, and generally preferable option, to the remove camera movement transform. The intent is the same. It is assumed that the camera that has recorded the motion of the particles in this particle record possesses some superimposed motion that needs to be removed. However, this transform caters for both x and y translations as well as rotation about the origin of the given coordinate system. The transformation information is stored in a vector 3D field 1D. The x and y components of this field correspond to the absolute x and y translations, while the z component corresponds to the absolute anti-clockwise rotation, in radians, around the origin. The rotation is applied first, followed by the translation.

Unlike the remove camera movement transform where the transformation information is incremental, this transform uses absolute translations and rotations. Thus no summation of previous movement is required.

While the user is at liberty to generate the transformation vector field in any manner they choose, two free processes, the transform from one path

field 2D process and the transform from two path fields 2D process, provide means for doing so from Lagrangian 2D path fields.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Use rotations* – If selected both translations and rotation will be included in the transform. If not the rotation information in the vector field will be ignored.
- *Field file name* – The path to the field file used in the transformation. This can be entered manually but generally is obtained by browsing the file system using the Browse button.
- *Browse…* – clicking this button allows the user to browse the file system and select the transformation vector field required.
- *Update movements* – clicking this button will load the date from the selected vector field and display it in the table in the Movements tab.

42 Transform Particle 3D Record Processes

42.1 Description

Particle 3D records can be transformed through a pipeline process that contains a number of individual transforms. The user can choose whether they wish to apply the transforms to a copy of the original particle record or to apply them to the original directly. The available transforms are listed below.

- *Linear transform*
- *Rotate transform*
- *Swap x transform*
- *Swap y transform*
- *Swap z transform*
- *Constant velocity transform*
- *Remove camera movement transform*

42.2 Views

Each transform pipeline has two views

- *Details view*
- *Transform view*

42.2.1 Details View

Description

The details view displays the process information, including its name, and whether a copy of the record should be used. There are two tabs:

Process

- *Process name* – the name the user wishes to give to the process.

Record

- *Create copy* – if selected the pipeline is applied to a copy of the record.
- *Particle record name* – if a copy is made then the new particle record will have this name.

Access

The details view is opened through the popup menu of the process object.

42.2.2 Transform view

Description

The transform view gives the user control over the transforms that comprise the pipeline. It is the default view. It contains the following elements:

- *Transforms defined for this pipeline* – this list box contains of the transforms currently in the pipeline. They are executed in the order in which they appear in the list.
- *New* – creates a new transform object and adds it to the list. A dialog box is displayed that prompts the user for the type of transform required.
- *Open* – loads a previously saved transform object from disk and adds it to the list. These transform files have *.ptr3d* extension.
- *Insert before…* – creates a new transform an inserts it before the transform selected in the list. If no transform is selected it will be added to the end of the pipeline.
- *Delete* – deletes a selected transforms object from the list. The user is prompted to confirm this action.
- *Clear* – this button clears all transforms from the list. The user is prompted to confirm this action.
- *Execute* – if the pipeline is associated with a particle record this button executes the pipeline. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the pipeline, or by double clicking the object.

42.3 Popup menu

The transform pipeline process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open transform view* – opens the object's transform view. This is the default.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

42.4 Transforms

In this section we describe the individual transforms that are available for transforming a particle record object.

42.4.1 Views

Each transform has a single view, the details view and the details that can be set in each of these views are given in subsequent sections.

42.4.2 Popup menus

The popup menu for each field transform has the following options:

- *Saves as…* – saves the transform to a disk file with a *.ptr3d* extension.
- *Open details view* – opens the details view.

42.4.3 Linear transform

This transform enables the particle 3D record spatial and temporal coordinates to be scaled and, in the case of the spatial coordinates, translated.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Spatial scaling* – all three spatial coordinates are multiplied by this factor.
- *x offset* – the x coordinate translation or offset.
- *y offset* – the y coordinate translation or offset.
- *z offset* – the z coordinate translation or offset.
- *Temporal scaling* – the time is multiplied by this factor.

42.4.4 Rotate transform

This transform rotates the physical domain of the particle record. Three rotations are executed in order. The first rotates about the x axis, the second about the ye axis and third about the z axis. The user can specify the rotation angles in each case.

The parameters are

- *x angle (deg)* – the angle of rotation (in degrees) about the x axis.
- *y angle (deg)* – the angle of rotation (in degrees) about the y axis.
- *z angle (deg)* – the angle of rotation (in degrees) about the z axis.

42.4.5 Swap x transform

This transform reflects the record in a vertical plane normal to the x axis and in the middle of the record. The x coordinates are swapped in each frame.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

42.4.6 Swap y transform

This transform reflects the record in a horizontal plane normal to the y axis and in the middle of the record. The y coordinates are swapped in each frame.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

42.4.7 Swap z transform

This transform reflects the record in a vertical plane normal to the z axis and in the middle of the record. The z coordinates are swapped in each frame.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

42.4.8 Constant velocity transform

This transform applies a constant velocity transformation to the frames within the particle record. Thus the coordinate origin of each frame is translated by a displacement given by the product of the specified velocity and the time since the start of the record. The velocity can be thought of as that of the camera recording the particles in the record.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x velocity* – the x component of the applied velocity.
- *y velocity* – the y component of the applied velocity.
- *z velocity* – the z component of the applied velocity.

42.4.9 Remove camera movement transform

This transform assumes that the camera used to capture the images in the original image sequence has a superimposed, unwanted motion perhaps due to vibration. It is also assumed that this camera motion has been recorded by all particles in the frame and thus by tracking any particle, stationary in the laboratory frame, the unwanted camera motion can be deduced. This camera motion must be stored in a vector 3D field 1D object (if x, y and z movement is to be removed) or a scalar 1D field object (if only x, y or z movement is to be removed). The transform will effectively subtract the recorded motion from all particles. Ideally the

field object should have the same number "frames" as the particle record being transformed.

The parameters are:

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Transform allows for* – four radio buttons are provided that allow the user to choose whether x, y and z movement is to be removed or only x, y or z movement. In the first case the vector 3D field 1D object must be provided for the transform while for the latter three a scalar 1D field is required.
- *Field file name* – the user can select the field file (vector or scalar field) using the Browse button.

43 Transform Eulerian Field Processes

43.1 Description

All Eulerian fields, both 2D and 3D, can be transformed by an associated transform pipeline process. This process can include a number of Eulerian field transforms, executed as a pipeline, and creates a new Eulerian field. There are eight different transforms that can be included in the pipeline, although not all of these are available for all Eulerian fields. For example, displacement fields do not support the constant velocity or field displacement transforms and 2D Eulerian fields don't support the swap z transform. These transforms are

- *Linear transform*
- *Swap x transform*
- *Swap y transform*
- *Swap z transform*
- *Subset transform*
- *Time average transform*
- *Constant velocity transform*
- *Field displacement transform*

43.2 Views

Each transform pipeline has two views

- *Details view*
- *Transform view*

43.2.1 Details View

Description

The details view displays the process information, including its name, and the name of the transformed field.

- *Process name* – the name the user wishes to give to the process.
- *Field name* – the name of the transformed field.

Access

The details view is opened through the popup menu of the process object.

43.2.2 Transform view

Description

The transform view gives the user control over the transforms that comprise the pipeline. It is the default view. The view includes the following elements:

- *Transforms defined for this pipeline* – this list box contains of the transforms currently in the pipeline. They are executed in the order in which they appear in the list.
- *New* – creates a new transform object and adds it to the list. A dialog box is displayed that prompts the user for the type of transform required.
- *Load* – loads a previously saved transform object from disk and adds it to the list. These transform files have extensions that depend on the field being transformed. For a velocity 2D field it is *.vtr2d*, for a velocity 3D field it is *.vtr3d*, for an intensity field it is *.itr*, for a density field *.dtr* and for a shape field *.str*.
- *Insert before…* – creates a new transform an inserts it before the transform selected in the list. If no transform is selected it will be added to the end of the pipeline.
- *Delete* – deletes a selected transforms object from the list. The user is prompted to confirm this action.
- *Clear* – this button clears all transforms from the list. The user is prompted to confirm this action.
- *Execute* – if the pipeline is associated with a field of the right type this button executes the pipeline. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the pipeline, or by double clicking the object.

43.3 Popup menu

The transform pipeline process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open transform view* – opens the object's transform view. This is the default.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

43.4 Transforms

In this section we describe the individual transforms that are available for transforming an Eulerian field time series object.

43.4.1 Views

Each transform has a single view, the details view and the details that can be set in each of these views are given in subsequent sections.

43.4.2 Popup menus

The popup menu for each field transform has the following options:

- *Saves as…* – saves the transform to a disk file with the appropriate field extension. For a velocity 2D field it is *.vtr2d*, for a velocity 3D field it is *.vtr3d*, for an intensity field it is *.itr*, for a velocity-intensity field it is .vit, for a density field *.dtr* and for a shape field *.str*.
- *Open details view* – opens the details view.

43.4.3 Linear transform

Under this transform every field value is multiplied by a scalar and a constant is added to it. For vector fields this is a constant vector, for scalar fields a scalar. In addition the spatial and temporal coordinates can be scaled and shifted. This transform is particularly useful for the non-dimensionalisation of the data.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Field scaling* – all values are multiplied by this factor.
- *Constant vector x comp* – the x component of the constant vector added to all vectors. Available for vector fields only.
- *Constant vector y comp* – the y component of the constant vector added to all vectors. Available for vector fields only.
- *Constant vector z comp* – the y component of the constant vector added to all vectors. Available for vector 3D fields only.
- *Constant field value* – the constant added to all field values. Available for scalar fields only.
- *Spatial scaling* – the spatial coordinates are multiplied by this scale factor.
- *x offset* – the x coordinate has this value added to it.
- *y offset* – the y coordinate has this value added to it.
- *z offset* – the z coordinate has this value added to it. Available only for 3D fields.
- *Temporal scaling* – the time scale is multiplied by this scale factor.
- *t offset* – time has this value added to it.

43.4.4 Swap x transform

This transform reflects the field in a vertical line or plane through the middle of the field. For a vector field all vectors have their x components reversed while for scalar fields the data is unchanged. The x coordinates are swapped in each frame. Scaling of both the field and spatial coordinates is possible.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Field scaling* – all field values are multiplied by this factor.
- *Spatial scaling* – the spatial coordinates are multiplied by this scale factor.

43.4.5 Swap y transform

This transform reflects the field in a horizontal line or plane through the middle of the field. For a vector field all vectors have their y components reversed while for scalar fields the data is unchanged. The y coordinates are swapped in each frame. Scaling of both the field and spatial coordinates is possible.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Field scaling* – all field values are multiplied by this factor.
- *Spatial scaling* – the spatial coordinates are multiplied by this scale factor.

43.4.6 Swap z transform

This transform, only available for 3D fields, reflects the field in a vertical plane through the middle of the field. For a vector field all vectors have their z components reversed while for scalar fields the data is unchanged. The z coordinates are swapped in each frame. Scaling of both the field and spatial coordinates is possible.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Field scaling* – all field values are multiplied by this factor.
- *Spatial scaling* – the spatial coordinates are multiplied by this scale factor.

43.4.7 Subset transform

This transform extracts a subset of a field time series. The user is able to select the range of each of the three or four independent variables that is to be included in the extracted field. This transform can be useful in eliminating extraneous data that is not needed in the field, thus saving processing time and memory space.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *x coordinate start* – this combo box allows the user to select the minimum value of the x coordinate in the transformed field time series.
- *x coordinate finish* – this combo box allows the user to select the maximum value of the x coordinate in the transformed field time series.
- *y coordinate start* – this combo box allows the user to select the minimum value of the y coordinate in the transformed field time series.
- *y coordinate finish* – this combo box allows the user to select the maximum value of the y coordinate in the transformed field time series.

- *z coordinate start* – this combo box allows the user to select the minimum value of the z coordinate in the transformed field time series. Available only for a 3D field.
- *z coordinate finish* – this combo box allows the user to select the maximum value of the z coordinate in the transformed field time series. Available only for a 3D field.
- *t coordinate start* – this combo box allows the user to select the start time for in the transformed field time series.
- *t coordinate finish* – this combo box allows the user to select the finial time in the transformed field time series.

43.4.8 Average fields in time transform

This transform takes groups of fields and averages them in time to produce a single field. The end result is a field time series with a reduced number of fields and a larger time step.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Average over frames* – the number of frames used in the averaging process.

43.4.9 Constant velocity transform

The fields in the time series are translated in space using a constant velocity vector. This transformation is useful if the user wishes to convert the data into a moving frame of reference. It is not available for displacement time series.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Reference time* – the field corresponding to this time will be used as the reference frame. This means that its spatial coordinates are invariant under the shift.
- *First time* – the first frame to which the transformation is applied.
- *Last time* – the last frame to which the transformation is applied.
- *x shift velocity* – the x component of the applied constant velocity field. This is equivalent to the camera moving to the right at this speed. Frames after the reference frame are shifted in the negative x direction when the shift velocity is positive, while those before are shifted to the right.

- *y shift velocity* – the y component of the applied constant velocity field. This is equivalent to the camera moving upwards at this speed. Frames after the reference frame are shifted in the negative y direction when the shift velocity is positive, while those before are shifted upwards.
- *z shift velocity* – the z component of the applied constant velocity field. This is equivalent to the camera moving outwards at this speed. Frames after the reference frame are shifted in the negative z direction when the shift velocity is positive, while those before are shifted in the positive z direction.

43.4.10 File displacements transform

This transform is only available for some 2D field time series. The fields in the time series are translated in space using displacement vectors stored in a text file. The number of vectors in the file (in which the x and y components are separated by spaces, as are the different vectors) should match the number of time frames in the field. If there are less vectors than expected the last vector is applied to the remaining frames in the sequence. If there are more vectors than expected, the excess vectors are ignored.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *File name* – the name of the file containing the displacement vectors.

44 Transform Lagrangian Path Fields Processes

44.1 Description

Transformation of Lagrangian path fields can be achieved through an associated transform pipeline process. This process can include a number of Lagrangian path field transforms, executed as a pipeline, and the result is a new Lagrangian path field. Currently only transform pipelines for 2D Lagrangian path fields are supported. There are three different transforms that can be included in the pipeline. These transforms are

- *Swap x*
- *Swap y*
- *Transform coordinates*

44.2 Views

Each transform pipeline has two views

- *Details view*
- *Transform view*

44.2.1 Details View

Description

The details view displays the process information, including its name, and whether a copy of the path field should be used. There are two tabs:

Process

- *Process name* – the name the user wishes to give to the process.

Path field

- *Create copy* – if selected the pipeline is applied to a copy of the path field.
- *Lagrangian path field name* – if a copy is made then the new path field will have this name.

Access

The details view is opened through the popup menu of the process object.

44.2.2 Transform view

Description

The transform view gives the user control over the transforms that comprise the pipeline. It is the default view. The view includes the following elements:

- *Transforms defined for this pipeline* – this list box contains of the transforms currently in the pipeline. They are executed in the order in which they appear in the list.
- *New* – creates a new transform object and adds it to the list. A dialog box is displayed that prompts the user for the type of transform required.
- *Load* – loads a previously saved transform object from disk and adds it to the list. These transform files have extensions of *.ltr2d*.
- *Insert before…* – creates a new transform an inserts it before the transform selected in the list. If no transform is selected it will be added to the end of the pipeline.
- *Delete* – deletes a selected transforms object from the list. The user is prompted to confirm this action.
- *Clear* – this button clears all transforms from the list. The user is prompted to confirm this action.
- *Execute* – if the pipeline is associated with a field of the right type this button executes the pipeline. If the pipeline is attached to some other type of entity, an image sequence for example, this button is disabled.

Access

This view is accessed through the popup menu of the pipeline, or by double clicking the object.

44.3 Popup menu

The transform pipeline process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.

- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open transform view* – opens the object's transform view. This is the default.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

44.4 Transforms

In this section we describe the individual transforms that are available for transforming a Lagrangian 2D path field.

44.4.1 Views

Each transform has a single view, the details view and the details that can be set in each of these views are given in subsequent sections.

44.4.2 Popup menus

The popup menu for each field transform has the following options:

- *Saves as…* – saves the transform to a disk file with the extension *.ltr2d*.
- *Open details view* – opens the details view.

44.4.3 Swap x transform

This transform changes the sign of the x coordinate of each particle.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

44.4.4 Swap y transform

This transform changes the sign of the y coordinate of each particle.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.

44.4.5 Transform coordinates

This transform matches the similarly named transform for the particle 2D record. It transforms the x and y coordinates of each particle using a time dependent transformation that includes both a rotation about the origin of the coordinate system and a translation in the x and y directions. The transformation information is stored in a vector 3D field 1D. The x and y components of this field correspond to the absolute x and y translations, while the z component corresponds to the absolute anti-clockwise rotation, in radians, around the origin. The rotation is applied first, followed by the translation.

While the user is at liberty to generate the transformation vector field in any manner they choose, two free processes, the transform from one path field 2D process and the transform from two path fields 2D process, provide means for doing so from Lagrangian 2D path fields.

The parameters are

- *Transform name* – a user specified name for the transform. The default name is the transform type.
- *Use rotations* – If selected both translations and rotation will be included in the transform. If not the rotation information in the vector field will be ignored.
- *Field file name* – The path to the field file used in the transformation. This can be entered manually but generally is obtained by browsing the file system using the Browse button.
- *Browse…* – clicking this button allows the user to browse the file system and select the transformation vector field required.
- *Update movements* – clicking this button will load the date from the selected vector field and display it in the table in the Movements tab.

45 Join Lagrangian Paths Process

45.1 Description

The join Lagrangian paths process enables the user to reconnect paths that have been broken for a variety of reasons. The break is most likely to have occurred because the particle being tracked has been lost from view, either because it has left the view or because the particle identification algorithm failed to identify it. No matter what the cause it can be useful to reestablish connections so that more complete paths are obtained.

The algorithm for seeking lost connections is relatively simplistic and is intended primarily for paths with small steps between frames. In fact the motivation for this process has come from tracking sediment motion with the aim of connecting a path that ends at a particular location with a path that starts at that same location some number of frames later. The assumption, of course, is that this is the same particle but the matching process has failed to find the complete path. The algorithm can be instructed to look for future paths that start within a specified region relative the end of the path being considered. Similarly the maximum time delay between the end of one path and the beginning of the other can be set. The user may also specify regions in which the search for connections will take place (if no regions are defined the whole space is considered) and whether the particle diameter should be used as a criterion for deciding whether a connection is valid or not. It is possible that more than one potential paths could be connected to the path under consideration. The user has the choice of whether the physical or temporal separations of the two paths should be most critical in determining the connection to be made. In other words paths that are closer together in time or space are preferred. In either case if more than one connection still remain as possible candidates the other criteria is used, and finally in both cases the diameter criteria is used if this is selected. When a potential connection has been found the user is able to specify how the missing portion of the path is constructed. It can be a linear interpolation from the end of the first path to the beginning of the second, leaving the particle at the end of the first path or leaving it at the beginning of the second for the duration of the gap between them.

45.2 Views

The process as a single view

- *Details view*

45.2.1 Details View

Description

The details view displays the process information, including its name, and the process parameters. These are

Process

- *Process name* – specifies the name of the process.

Parameters

- *Maximum frames offset* – the maximum number of frames between the end of the first path and beginning of the second for consideration as a connection. A value of one implies that the start and end occur in adjacent frames.
- *Left* – the position of the left hand side of the search window relative to the end of the first path. Only paths that start within this search window will be considered for joining to the end of the first path.
- *Bottom* – the position of the bottom of the search window relative to the end of the first path.
- *Back* – (for Lagrangian 3D path fields only) the position of the back of the search window relative to the end of the first path.
- *Width* – the width of the search window.
- *Height* – the height of the search window.
- *Depth* – (for Lagrangian 3D path fields only) the height of the search window.
- *Use particle diameter* – if selected the diameter of the particles at the end of the first path and beginning of the second path will be considered in determining whether the paths should be joined. In this case the maximum diameter change parameter specifies the fraction this diameter can change.
- *Maximum diameter change* – if the particle diameter is to be used to determine whether two paths should be joined this parameter specifies the fraction the diameter can change before the join is excluded. Thus a figure of 0.2 states that the diameter of the particle at the beginning of the second path can be between 80% and 120% of the diameter of the particle at the end of the first path.

- *Interpolate position* – if selected the position of the particle in intervening frames between the end of the first path and beginning of the second will be linearly interpolated. If the end and start frames are subsequent this has no effect.
- *Use start position* – if selected the position of the particle in intervening frames between the end of the first path and beginning of the second will be set to the position at the end of the first path. If the end and start frames are subsequent this has no effect.
- *Use start position* – if selected the position of the particle in intervening frames between the end of the first path and beginning of the second will be set to the position at the end of the first path. If the end and start frames are subsequent this has no effect.
- *Start with closest in time* – if selected, and more than one candidate path lies within the search window, the path that starts soonest after the end of the path being considered is selected. If more than one path starts at this minimum time separation the path that starts at the closest point in space to the end of the path under consideration is selected. Again if more than one path is equidistant from the end point then the path whose particle diameter most closely matches that at the end of the path under consideration is chosen, if the user has chosen to use this as a criterion.
- *Start with closest in space* – if selected, and more than one candidate path lies within the search window, the path that starts at the closest point in space to the end of the path under consideration is selected. If more than one path is equidistant from the end of the path under consideration then the path that starts soonest after the end of the path being considered is selected. Again if more than one path starts at this minimum time separation, the path whose particle diameter most closely matches that at the end of the path under consideration is chosen, if the user has chosen to use this as a criterion.

Regions

- *Analysis regions* – the list of regions (2D regions for Lagrangian 2D path fields and 3D regions for Lagrangian 3D path fields) that are used in the process. Only paths that end and start in these regions will be considered for connection.
- *Load* – enables the user to add a 2D or 3D region(s) from a disk file to the analysis regions.
- *Delete* – deletes the selected region from the analysis regions list.

Access

The details view is opened through the popup menu of the process object or by double clicking on it.

45.3 Popup menu

The join Lagrangian paths process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the processes currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

46 Save Entity Processes

46.1 Description

A number of processes are available that play the simple role of saving an entity to disk. These processes are normally only used within a process pipeline to enable an intermediate result entity to be archived. For example consider a process pipeline that takes an image sequence, identifies the particles within the images, matches the particles with a PTV 2D analysis pipeline and creates a velocity 2D field from the particle 2D record. The outcome of this pipeline is a velocity field while the particle record created as part of the process is discarded. A save particle 2D record process inserted into the pipeline at an appropriate place would enable this record to be saved for later analysis.

Save processes are provided for:

- *Image sequences*
- *Particle 2D records*
- *Particle 3D records*
- *Particle 2D record bundles*
- *Intensity field time series*
- *Velocity 2D field time series*
- *Velocity 3D field time series*
- *Velocity-intensity field time series*
- *Density field time series*
- *Shape field time series*
- *Displacement 2D field time series*
- *Displacement 3D field time series*
- *Displacement 3D surface field time series*
- *Lagrangian 2D path fields*
- *Lagrangian 3D path fields*

All save processes take entities of the appropriate type as inputs and provide the same entity as their output.

46.2 Views

Each save process provides a single view known as the details view.

46.2.1 Details View

Description

The details view displays the process information, including its name, and the name of the transformed field.

- *Process name* – the name the user wishes to give to the process.
- *Entity file name* – the name of the file in which the entity is to be saved.
- *Close entity if last process* – if this box is checked the entity which is saved by this process is also closed so that it doesn't appear in the *Streams* system. For pipelines executed using a batch process this box should be ticked to ensure that the system does not become starved of memory.

Access

The details view is opened through the popup menu of the process object.

46.3 Popup menu

A save entity process provides the following options in its popup menu.

- *Save as…* – saves the process in a disk file. All bound processes have the extension *.pcb*.
- *Lock* – locks the process.
- *Change name…* – opens a dialog box that enables the user to change the name of the process.
- *Open details view* – opens the object's details view.
- *Execute* – executes the process as a standalone process. If the associated entity is not of the appropriate type the execution will fail.
- *Close all views* – closes all of the currently open views.
- *Close* – currently this too closes all views. To remove the object from a process view the user should use the delete button.

47 Regions in 2D

47.1 Description

Two-dimensional regions are simply geometric shapes defined in two-dimensional space that can be used in a variety of settings within *Streams*. Regions can be created within a number of entity views, for example the particle 2D record particle view, the image sequence image view, and some of the field views described in the *User's Guide: Field Objects*. Six different region shapes can be created – points, lines, rectangles, circles, polygons and freehand figures (that are essentially many-sided polygons). Regions, other than the point and line regions, have an additional property. They either define the space internal to the shape, or the space external to it. Thus an internal rectangle defines the space within the four sides of the rectangle, while an external rectangle includes all space excluding that which lies within its four sides. The colour of the region's outline distinguishes between the internal and external setting. Internal regions are drawn in blue or green, and external regions are drawn in red or orange.

In many applications regions can be selected by double clicking inside the region (or very close to the region in the case of a point or line) that is to be selected. Note that *inside* means inside the figure, independent of whether the region is an internal or external region. Many regions can be selected at the same time using the same process. Regions are unselected in the same way.

Selected regions can be used for different purposes. For example in the particle view of a particle 2D record the user can delete particles in the currently selected regions. Selected and unselected regions are differentiated by their colour. Selected internal regions are coloured blue, unselected internal regions are coloured green, selected external regions are coloured red, while unselected external regions are coloured orange.

The user can manipulate selected regions. Selected regions are moved by holding down the CTRL key and using the four arrow keys. They can be resized by holding the ALT key and using the UP and DOWN arrow keys. For these manipulations it is important that the containing view has the mouse and keyboard focus. These are obtained by clicking the left mouse button on the view.

Regions are drawn and manipulated in the same way, independent of the view in which they appear.

Regions are drawn in the following way.

- *Point* – the mouse is moved to the location of the point and the SHIFT key is held down while the left mouse button is clicked.
- *Line* – the left mouse button and SHIFT key are pressed when the cursor is located at one end of the line. The mouse is dragged (i.e. left button pressed) to the other end of the line and the button released. A rubber band is drawn during the dragging action. The SHIFT key need not be held during the drag process.
- *Rectangle* – the left mouse button and SHIFT key are pressed when the cursor is located at one corner of the rectangle. The mouse is dragged (i.e. left button pressed) to the opposite corner of the rectangle and the button released. A rubber band is drawn during the dragging action. The SHIFT key need not be held during the drag process.
- *Circle* – the left mouse button and SHIFT key are pressed when the cursor is located at the centre of the circle. The mouse is dragged (ie left button pressed) to the circumference of the circle and the button released. A rubber band is drawn during the dragging action. The SHIFT key need not be held during the drag process.
- *Polygon* – the SHIFT key is held down and the left mouse button is clicked (not held down), at each vertex of the polygon and the mouse moved to the next vertex, where again the mouse is clicked. The polygon is completed by clicking the mouse very close to the starting vertex (care needs to be taken to make this as accurate as possible). The ESCAPE key can be pressed at any time during this action in order to abort the process. A rubber band is drawn during the movement of the mouse.
- *Freehand* – the left mouse button and SHIFT key are pressed. A freehand figure is drawn as the mouse is dragged. When the mouse button is released the figure is completed with a straight line drawn from the final point to the initial point. The SHIFT key need not be held during the drag process.

Two-dimensional regions can be stored on disk in *.rgn2* files and loaded later when needed. These files can contain one or more regions.

Some examples of where 2D regions are used follow.

Image sequence image view

In the image regions are created for later use such as in the definition of some image filters or in particle identification objects.

Particle 2D record particle view

In the particle 2D record particle view regions are useful for eliminating erroneous or undesirable particles from the frames within the record. They are also useful in allowing the user to explore details of the matching process through the match details view.

Overwrite regions filter

The overwrite regions filter provides the user with a mechanism for altering the pixel information in selected regions within each image in the sequence. Regions are used to specify the regions that are to be manipulated.

Eulerian field creation

Regions can be used to exclude certain regions from the calculation of the grid-based Eulerian fields. For example a physical space within a fluid flow, that is not actually part of the flow, can be excluded from the production of the time series. The flow around a body is an example of such a situation.

PTV 2D analysis

Regions can be used as a way of limiting the particles in a frame that can participate in the matching process. This is a powerful technique that enables the user to analyse different parts of the flow using different analyses. The reason for doing this may be that different parts of the flow have different characteristics that demand different costing strategies, or different optimisation parameters.

Field views

Regions are used in a number of ways in the graphical views of scalar and vector fields. The most common use is for defining a zoom rectangle that allows the user to enlarge a portion of the view. The selection, movement and enlargement of zoom rectangles are not supported in these views.

47.2 Views

Each type of region provides a details view in which the user can manually edit the region's parameters. These are described in the next subsections.

47.2.1 Point Details View

- *Region name* – a user defined name for the region.
- *x, y* – the x, y coordinates of the point.
- *Selected* – this box is checked when the region is selected.

47.2.2 Line Details View

- *Region name* – a user defined name for the region.
- *Start x, y* – the x, y coordinates of the start of the line.
- *End x, y* – the x, y coordinates of the end of the line.
- *Selected* – this box is checked when the region is selected.

47.2.3 Rectangle Details View

- *Region name* – a user defined name for the region.
- *Left x* – the x coordinate of the left of the rectangle.
- *Right x* – the x coordinate of the right of the rectangle.
- *Bottom y* – the y coordinate of the bottom of the rectangle.
- *Top y* – the y coordinate of the top of the rectangle.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

47.2.4 Circle Details View

- *Region name* – a user defined name for the region.
- *Centre x, y* – the x, y coordinates of the centre of the circle.
- *Radius* – the radius of the circle.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

47.2.5 Polygon Details View

- *Region name* – a user defined name for the region.
- *Current vertices* – a list of the vertices, in the order they were drawn or created, that define the polygon. Each vertex is defined by its (x,y) coordinates.
- *Move up* – moves the selected vertices in the list up one place.
- *Move down* – moves the selected vertices in the list down one place.
- *Delete vertices* – deletes all selected vertices in the list. The user must confirm this action.

- *New vertex x* – the x coordinate of a new vertex to be added or inserted in the list.
- *New vertex y* – the y coordinate of a new vertex to be added or inserted in the list.
- *Add vertex* – adds a new vertex to the list using the coordinates specified in the New vertex x and y fields.
- *Insert vertex* – inserts a new vertex into the list before the currently selected vertex. The coordinates of the new vertex are specified in the New vertex x and y fields.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

47.2.6 Freehand Details View

The freehand region is the only region which cannot be edited manually except for its name and whether it is internal or selected.

- *Region name* – a user defined name for the region.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

47.3 Popup menu

Regions may have popup menus, depending on the view that employs them. In the particle record particle view popup menus are displayed by right clicking the mouse within the region.

47.4 Region 2D manager view

All views that support two-dimensional regions provide access to a region 2D manager view. This view provides access to the regions defined for the view, enabling new regions to be created, and existing regions to be edited (through their details view) or deleted. The manager view has the following elements:

- *Current regions* – this list box displays all of the currently defined regions for the view – using a simple icon and the region name. This list box supports drag and drop of 2D regions.
- *Load…* – this button enables the user to load previously saved regions from a file. The file extension is *.rgn* or *.rgn2*.
- *Save…* – this button saves all currently defined regions to a file with a *.rgn2* extension.

- *Delete…* deletes all selected regions in the list. The user is asked to confirm the action.
- *Clear…* – deletes all regions in the list. The user is asked to confirm the action.

Menu

The manager view has a single menu entitled Region. It has the options that allow the user to create regions manually, with the exception of freehand regions. Freehand regions can only be created through drawing them on screen.

- *Load…* – enables the user to load previously saved regions from a file. The file extension is *.rgn2*.
- *Load from text file…*– loads 2D regions from a text file. See section 47.5.
- *Create line…* – creates a new line and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create circle…* – creates a new circle and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create rectangle…* – creates a new rectangle and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create point…* – creates a new point and adds it to the current regions list and displays the details view so that its parameters can be editted..
- *Create polygon…* – creates a new polygon and adds it to the current regions list and displays the details view so that its parameters can be editted.

47.5 Loading regions from text file

The region 2D manager view enables the user to create new regions by extracting region parameters from a text file. Sometimes it is simpler, and more accurate, to create a set of regions within a spreadsheet and then export these to *Streams*. The file should be a space delimited file (*.prn* if using Excel©) and have the following format.

The first line of the file should contain a single integer value specifying the number of regions to be read from the file. There should follow one line per region. This line should have data in the following order:

- *Region code* – this is an integer that specifies the type of region about to be defined. The codes are rectangle 1, polygon 2, circle 3, freehand 4, point 5, line 6.
- *Parameters that define the region* – a series of data values unique to each region type. See below for details.

- *Internal* – a value of 1 if the region is an internal region, and 0 if an external region. Note that this parameter is not relevant to points and lines.
- *Selected* – value of 1 if the region is selected and a value of 0 if it is not.
- *Name* – the name of the region. This must be a string without spaces and not enclosed within quotes.

Each of the regions has region specific data that must be included in the file as a series of numbers. The following sections describe the data for each region type.

Rectangle

- *x max* – the maximum value of x.
- *y max* – the maximum value of y.
- *x min* – the minimum value of x.
- *y min* – the minimum value of y.

Polygon

- number of vertices – the number of vertices in the polygon. For each vertex there is a pair of x y coordinates as given below.
- *x* – the x coordinate of the next vertex.
- *y* – the y coordinate of the next vertex.

Circle

- *radius* – the radius of the circle.
- *centre x* – the x coordinate of the circle's centre.
- *centre y* – the y coordinate of the circle's centre.

Freehand

- number of vertices – the number of vertices in the freehand polygon. For each vertex there is a pair of x y coordinates as given below.
- *x* – the x coordinate of the next vertex.
- *y* – the y coordinate of the next vertex.

Point

- *x* – the x coordinate of the point.
- *y* – the y coordinate of the point.

Line

- *x1* – the x coordinate of the start point.
- *y1* – the y coordinate of the start point.
- *x2* – the x coordinate of the end point.

- *y2* – the y coordinate of the end point.

48 Regions in 3D

48.1 Description

Three-dimensional regions are three dimensional objects defined in three-dimensional space. The share many of the same characteristics as 2D regions described in chapter 45. For example regions that can be internal or external and selected or not selected, and they use the same colours to differentiate between these different states.

The primary difference between 2D and 3D regions, other than the obvious difference in the number of dimensions, is that 3D regions cannot be drawn or manipulated on screen. Instead they must either be crafted manually through the settings in the region's details view, or by reading the parameters defining the region from a text file. This latter option will be discussed in section 48.5. Thus 3D regions cannot be moved on screen using the mouse and keyboard commands, nor can they be selected or deselected with the mouse.

Seven different 3D regions are provided. These are:

- *Point*
- *Line*
- *Sphere*
- *Cylinder*
- *Cone*
- *Rectangle*
- *General rectangle*

The meaning of the first five of these is obvious, the latter two perhaps less so. A rectangle is a rectangular prism whose sides are parallel to the three coordinate axes. A general rectangle is a rectangular prism that can have any orientation in space.

Three-dimensional regions can be stored on disk in *.rgn3* files and loaded later when needed. These files can contain one or more regions.

48.2 Views

Each type of region provides a details view in which the user can manually edit the region's parameters. These are described in the next subsections.

48.2.1 Point Details View

- *Region name* – a user defined name for the region.
- *x, y, z* – the x, y and z coordinates of the point.
- *Line width* – points in 3D can be difficult to see depending on where they are located. The line width provides a mechanism by which the point can be slightly enlarged to make it visible.
- *Inside* – this box is checked when the region is internal. For a point this has little meaning.
- *Selected* – this box is checked when the region is selected.

48.2.2 Line Details View

- *Region name* – a user defined name for the region.
- *Start x, y, z* – the x, y and z coordinates of the start of the line.
  End x, y, z – the x, y and z coordinates of the end of the line.
- *Inside* – this box is checked when the region is internal. For a line this has little meaning.
- *Selected* – this box is checked when the region is selected.

48.2.3 Sphere Details View

- *Region name* – a user defined name for the region.
- *Centre x, y, z* – the x, y, z coordinates of the centre of the sphere.
- *Radius* – the radius of the sphere.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

48.2.4 Cylinder Details View

A cylinder is defined by a point that lies in the centre of its base, a vector parallel to the axis of the cylinder, a radius of the cylinder and its length. The axis is a unit vector but the user can enter a non-unit vector as *Streams* will scale it appropriately.

- *Region name* – a user defined name for the region.
- *Base centre x, y, z* – the x, y and z coordinates of the centre of the base of the cylinder.
- *Axis x, y, z* – the x, y and z components of the cylinder axis.
- *Radius* – the radius of the cylinder.
- *Length* – the length of the cylinder.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

48.2.5 Cone Details View

A cone is defined by a point at its tip, a vector parallel to the axis of the cone (from tip to base), a radius of the base and a length. The axis is a unit vector but the user can enter a non-unit vector as *Streams* will scale it appropriately.

- *Region name* – a user defined name for the region.
- *Tip x, y, z* – the x, y and z coordinates of the tip of the cone.
- *Axis x, y, z* – the x, y and z components of the cone axis.
- *Radius* – the radius of the cone base.
- *Length* – the length of the cone.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

48.2.6 Rectangle Details View

- *Region name* – a user defined name for the region.
- *Left x* – the left of the rectangular prism.
- *Right x* – the right of the rectangular prism.
- *Bottom y* – the bottom of the rectangular prism.
- *Top y* – the top of the rectangular prism.
- *Back z* – the back of the rectangular prism.
- *Front z* – the front of the rectangular prism.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

48.2.7 General Rectangle Details View

A general rectangular prism is defined by a point at its centre, two normal axes, nominally called the width and height axes, and its width, height, and depth. The prism extends either side of the centre a distance

width/2 along the width axis, a distance height/2 along the height axis, and a distance depth/2 along the axis obtained by taking the cross-product of the width and height axes. Both axes are unit vectors but the user can enter them as non-unit vectors as *Streams* will scale them appropriately.

- *Region name* – a user defined name for the region.
- *Centre x, y, z* – the x, y and z coordinates of the centre of the prism.
- *Width axis x, y, z* – the x, y and z components of the width axis.
- *Height axis x, y, z* – the x, y and z components of the height axis.
- *Width* – the width of the prism in the width axis direction.
- *Height* – the height of the prism in the height axis direction.
- *Depth* – the depth of the prism in the depth axis (obtained from the cross-product of the width and height axes) direction.
- *Inside* – this box is checked when the region is internal.
- *Selected* – this box is checked when the region is selected.

48.3 Popup menu

Region popup menus are accessed through the objects listed in the region 3D manager view – see next section.

48.4 Region 3D manager view

All views that support three-dimensional regions provide access to a region 3D manager view. This view provides access to the regions defined for the view, enabling new regions to be created, and existing regions to be edited (through their details view) or deleted. The manager view has the following elements:

- *Current regions* – this list box displays all of the currently defined regions for the view – using a simple icon and the region name. This list box supports drag and drop of 3D regions.
- *Load…* – this button enables the user to load previously saved regions from a file. The file extension is *.rgn3*.
- *Save…* – this button saves all currently defined regions to a file with a *.rgn3* extension.
- *Delete…* deletes all selected regions in the list. The user is asked to confirm the action.
- *Clear…* – deletes all regions in the list. The user is asked to confirm the action.

Menu

The manager view has a single menu entitled Region. It has the options that allow the user to create regions manually.

- *Load…* – enables the user to load previously saved regions from a file. The file extension is *.rgn3*.
- *Load from text file…*– loads 3D regions from a text file. See section 48.5.
- *Create point…* – creates a new point and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create line…* – creates a new line and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create cylinder…* – creates a new cylinder and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create cone…* – creates a new cone and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create sphere…* – creates a new sphere and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create rectangle…* – creates a new rectangular prism and adds it to the current regions list and displays the details view so that its parameters can be editted.
- *Create general rectangle…* – creates a new general rectangular prism and adds it to the current regions list. The prism will have default parameters that will need to be editted.

48.5 Loading regions from text file

The region 3D manager view enables the user to create new regions by extracting region parameters from a text file. Sometimes it is simpler, and more accurate, to create a set of regions within a spreadsheet and then export these to *Streams*. The file should be a space delimited file (*.prn* if using Excel©) and have the following format.

The first line of the file should contain a single integer value specifying the number of regions to be read from the file. There should follow one line per region. This line should have data in the following order:

- *Region code* – this is an integer that specifies the type of region about to be defined. The codes are rectangle 11, sphere 12, cylinder 13, point 14, line 15, general rectangle 16, cone 19.
- *Colour code* – as the colour of regions is determined by whether they are selected or internal/external no colour information is required. Thus the colour code is -1. All 3D regions require this value.

- *Parameters that define the region* – a series of data values unique to each region type. See below for details.
- *Internal* – a value of 1 if the region is an internal region, and 0 if an external region. Note that this parameter is not relevant to points and lines.
- *Selected* – value of 1 if the region is selected and a value of 0 if it is not.
- *Name* – the name of the region. This must be a string without spaces and not enclosed within quotes.

Each of the regions has region specific data that must be included in the file as a series of numbers. The following sections describe the data for each region type.

Rectangle

- *x origin* – the x coordinate of the origin of the rectangle (i.e. its left, bottom, back point).
- *y origin* – the y coordinate of the origin of the rectangle (i.e. its left, bottom, back point).
- *z origin* – the z coordinate of the origin of the rectangle (i.e. its left, bottom, back point).
- *width* – the width of the rectangle - i.e. the length in the x direction.
- *height* – the height of the rectangle - i.e. the length in the y direction.
- *depth* – the depth of the rectangle - i.e. the length in the z direction.

Sphere

- *radius* – the radius of the sphere.
- *centre x* – the x coordinate of the sphere's centre.
- *centre y* – the y coordinate of the sphere's centre.
- *centre z* – the z coordinate of the sphere's centre.

Cylinder

- *radius* – the radius of the cylinder.
- *length* – the length of the cylinder.
- *axis x* – the x component of the cylinder's axis.
- *axis y* – the y component of the cylinder's axis.
- *axis z* – the z component of the cylinder's axis.
- *base x* – the x coordinate of the centre of the base of the cylinder.
- *base y* – the y coordinate of the centre of the base of the cylinder.
- *base z* – the z coordinate of the centre of the base of the cylinder.

Point

- *x* – the x coordinate of the point.

- *y* – the y coordinate of the point.
- *z* – the z coordinate of the point.

Line

- *x1* – the x coordinate of the start point.
- *y1* – the y coordinate of the start point.
- *z1* – the z coordinate of the start point.
- *x2* – the x coordinate of the end point.
- *y2* – the y coordinate of the end point.
- *z2* – the z coordinate of the end point.

General rectangle

- *width* – the width of the rectangle - i.e. the length in the width direction.
- *height* – the height of the rectangle - i.e. the length in the height direction.
- *depth* – the depth of the rectangle - i.e. the length in the depth direction.
- *width axis x* – the x component of the rectangular prism's width axis.
- *width axis y* – the y component of the rectangular prism's width axis.
- *width axis z* – the z component of the rectangular prism's width axis.
- *height axis x* – the x component of the rectangular prism's height axis.
- *height axis y* – the y component of the rectangular prism's height axis.
- *height axis z* – the z component of the rectangular prism's height axis.
- *centre x* – the x coordinate of the sphere's centre.
- *centre y* – the y coordinate of the sphere's centre.
- *centre z* – the z coordinate of the sphere's centre.

Cone

- *radius* – the radius of the cone.
- *length* – the length of the cone.
- *axis x* – the x component of the cone's axis.
- *axis y* – the y component of the cone's axis.
- *axis z* – the z component of the cone's axis.
- *tip x* – the x coordinate of the tip of the cone.
- *tip y* – the y coordinate of the tip of the cone.
- *tip z* – the z coordinate of the tip of the cone.

All axis vectors are converted to unit vectors if they are not already.

49 Formulae

49.1 Description

Formulae provide a powerful tool for developing user-defined tools in a number of areas within *Streams*. In a general sense formulae are simply mathematical expressions that compute scalar quantities as functions of one or more variables. Specifically they can appear in three applications:

- *General mathematical formulae* – these formulae appear in areas that are not associated with Eulerian and Lagrangian field calculators. The formulae are based on a number of independent variables such as x, y, perhaps z, and t
- *Field calculator formulae*– these formulae are specifically adapted to the computation of fields based on Eulerian and Lagrangian fields. In this case the formulae can include the independent variables x, y, and t as well as field variables such as a velocity component or intensity field value.
- *Tablet formulae*– these formulae are specifically adapted to the manipulation of fields in tablets – see the *User's Guide: Tablet Objects* manual.

All of the formulae, whether they are of a general nature or specifically associated with field calculators or tablets, have a number of things in common. All formulae are defined by text typed into a text area provided in some dialog box or window. The text must make mathematical sense and will always be checked for consistency before the formula is accepted. A popup menu is always provided that lists the acceptable formula components. The popup menu is displayed by right clicking on the text area into which the formula is entered. The user is not obliged to use the popup menu to select elements for the formula but they may find it to be a useful prompt.

One area where these formulae do differ is in the presence, or not, of an equals sign (or assignment operator). Only for tablet formulae is this used.

All formulae can include the following common elements.

Unary operators

The standard unary operators are provided:

- – negative

- + positive

Binary operators

The following binary operators are provided:

- – subtract.
- + add
- * multiply
- / divide
- ^ exponent (power)

Constants

The following constants are provided:

- pi or PI

Mathematical functions

The following functions are provided:

- abs() – absolute value
- sin() – sine
- cos() – cosine
- tan() – tangent
- asin() – inverse sine or arc sine
- acos() – inverse cosine or arc cosine
- atan() – inverse tangent or arc tangent. The result always lies between $-\pi/2$ and $\pi/2$.
- exp() – exponential
- log() – natural logarithm
- sinh() – hyperbolic sine
- cosh() – hyperbolic cosine
- tanh() – hyperbolic tangent
- sqrt() – square root
- cbrt() – cube root
- h() – Heaviside function. This has a value of 1 if its argument is non-negative and a value of 0 if its argument is negative
- random() – Pseudo-random number generator. It generates a pseudo-random number between 0 and 1 each time it is executed.

Brackets

The following brackets are provided:

- ()
- []
- {}

For all formulae, except those in tablets, all of these brackets behave in the same way and can be used for grouping formula elements. The first of these is also used for the identification of arguments for mathematical functions. The formulae format should be natural to a mathematician. Blank spaces and new lines can appear anywhere in the formula, although not within individual tokens such as function names, but the formula is case sensitive so, SIN() cannot be used instead of sin().

In tablet formulae the curly and square brackets have special meanings. See the *User's Guide: Tablet Objects* manual.

49.2 General mathematical formulae

General mathematical formulae also include another set of formula elements. These are the independent variables that can be included in the mathematical expressions. These are not always the same and can always be found in the popup menu. For example, mathematical formulae are used in the custom theoretical model used for generating artificial particle 2D records. In this case the two velocity components, each with its own formula, will depend on the location of a particle in space and time. Thus x, y and t are the three independent variables available. On the other hand the mathematical formula filter computes the intensity of a pixel in an RGB image from the intensities of the red, green and blue guns. Thus the independent variables in this case are R, G and B – representing the values of the three colour channels – as well as the spatial coordinates x and y and the temporal coordinate t.

49.3 Field calculator formulae

The mathematical formulae used in the definition of custom field calculators include a number of other components.

Independent variables

For Eulerian field calculators the independent variables are the variables associated with the three or four-dimensional fields on which the calculator is operating – x, y, z (perhaps) and t. For a Lagrangian path field there are only 2 such variables, and in this case they are p and t, the path number and time.

Field variables

Unlike general mathematical formulae field calculator formulae have another set of independent variables, based on the field itself. In the case of a velocity 2D field time series the velocity components, u and v, their fluctuating components, u' and v', the turbulent stress, u'v', and so on can be included in the formulae. In fact it is much more likely that these variables will be used than the x, y and t variables.

Mathematical functions

Field calculator formulae normally provide a number of additional mathematical functions. The availability of these will depend on whether the Eulerian field has 3 or 4 independent variables. They are

- aveX() – averages its argument over x
- aveY() – averages its argument over y
- aveZ() – averages its argument over y
- aveT() – averages its argument over t
- aveXY() – averages its argument over both x and y
- aveXZ() – averages its argument over both x and z
- aveYZ() – averages its argument over both y and z
- aveXYT() – averages its argument over x, y and t.
- aveXYZ() – averages its argument over x, y, and z.
- aveXYZT() – averages its argument over x, y, z and t.

Lagrangian path field calculators provide different functions

- aveP() – averages its argument over all paths
- aveT() – averages its argument over time along each path
- avePT() – averages its argument over all paths and all time
- corr() – computes a normalised auto-correlation of its argument along each path. The resulting function interprets the t variable as the time offset in the auto-correlation. See the *System Theory and Design* manual.

Brackets

All field calculator based formulae provide an additional bracket, <>. These brackets are equivalent to a time average.

49.4 Tablet formulae

The tablet formulae are discussed in detail in the *User's Guide: Tablet Objects* manual.

50 Manuals

50.1 Description

Streams does not provide an on-line help system. Instead four extensive manuals, in PDF format, are available with the software. While these manuals can be readily accessed through a standard PDF reader, *Streams* also provides the capability of viewing the manuals within the application itself.

The user can create bookmarks within each of the manuals and these bookmarks can be saved to file and reinstated when the application is opened once again, and the manual loaded.

50.2 Views

Manuals provide two views.

- Details view
- Reader view

50.2.1 Details View

The details view displays the name of the manual and the number of pages it contains. The name of the manual is set to the name of the PDF file (without the extension) when it is first loaded, but the user is at liberty to change this name if they desire.

50.2.2 Reader View

The reader view is the default view for a manual. It displays the contents of the PDF file and allows the reader to navigate within the file in a number of different ways. One of the primary mechanisms provided for navigation is the bookmark. Any page within the manual can be marked. The user is prompted to provide a meaningful name for the bookmark to help remind them of its purpose. Bookmarks are stored with the manual while it is open in *Streams*, and, on closure, the user is at liberty to save the bookmarks to a file. These bookmarks can be reinstated when the manual is loaded again at a later time.

The view displays a single page of the manual at any one time. Scrollbars are provide if the page is not completely visible with the view. Above the area in which the page is rendered lie two toolbars, that provide navigation and general view control.

Navigation toolbar

The top toolbar is the navigation toolbar. It provides a number of buttons and control that facilitate navigation within the manual.

- ▶| – go to the next page.
- |◀ – go to the previous page.
- |▶ – go to the last page.
- ◀| – go to the first page.
- *Page number* – displays the current page number. By changing this value the user can go directly to a particular page.
- *Scroll pages* – the user can scroll through the manual using the scroll bar provided. In doing so the page number automatically updates.

Bookmark and zoom toolbar

The bottom toolbar, the bookmark and zoom toolbar, enables the user to create a bookmark and navigate within the document using any of the current defined bookmarks. It also allows the user to increase or decrease the size of the text by zooming in or zooming out.

- *Add bookmark* – allows the user to create a new bookmark. Each bookmark stores two pieces of information. The first is the page number, corresponding to the page currently displayed, and the second is a bookmark name, a simple piece of text, that describes the information that the bookmark is identifying. The user is prompted to enter the second of these in a dialog box.
- *Bookmarks* – this dropdown box lists all of the currently defined bookmarks for this manual. The name, followed by the page number, is displayed for each bookmark. By selecting a bookmark from the dropdown box the user is taken directly to the corresponding page in the manual. Bookmarks can be deleted through the bookmark manager view – see section 50.2.2.1.
- + – clicking this button zooms in so that the text becomes larger. Each click increases the text size by 5%.
- - – clicking on this button zooms out so that the text becomes smaller. Each click decreases the text size by 5%.

Access

This view is accessed through the popup menu of a manual or by double clicking the manual.

Menus

Bookmark Menu

- *Open bookmark manager view…* – this menu item opens the bookmark manager view. This view is described in section 50.2.2.1.

50.2.2.1 Bookmark manager view

The bookmark manager view allows the user to manage the bookmarks defined for this particular manual. The currently defined bookmarks are itemized in a list box, below which are a number of buttons. These are as follows:

- Load… – loads a set of bookmarks from a file and adds these bookmarks to those already appearing in the list. Bookmarks are stored in a file with a .bkmk extension.
- *Save…* – saves the currently defined bookmarks in a file with extension .bkmk.
- *Delete…*– deletes all of the selected bookmarks in the list. The user is asked to confirm this action.
- *Clear…*– deletes all of the bookmarks in the list. The user is asked to confirm this action.

50.3 Popup menu

The popup menu for a manual provides the following options:

- *Save bookmarks…* – saves the currently defined bookmarks in a file with extension .bkmk.
- *Load bookmarks…* – loads a set of bookmarks from a file and adds these bookmarks to those already defined for the manual. Bookmarks are stored in a file with a .bkmk extension.
- *Change name…* – opens a dialog box that enables the user to change the name of the manual.
- *Open details view* – opens the manual's details view.
- *Open reader view* – opens the manual's reader view.
- *Close all views* – closes all of the currently open views.

- *Close* – closes all views for the manual and removes the manual from the *Streams* system.