PTV manual

1 Introduction

Particle tracking velocimetry (PTV) is an experimental method to obtain information of the fluid velocity by tracking tracer particles in the flow. In short, a light source illuminates tracer particles in the flow, where a camera is used to record the coordinates of the tracer particles within the measurement domain. The images can then be used to determine the trajectories of the tracer particles and hence the velocity of the tracer in time. Typically, velocity data in the Lagrangian framework are converted to Eulerian velocity fields. Off the shelf PTV systems generally include all the necessities to go from experiments to velocity fields, such as a laser, lenses, mirrors, camera and software. While these systems are relatively easy to implement in the laboratory, they are very expensive. Designing your own system can therefore be beneficial and gives the user the flexibility to adapt the system where necessary. Developing a PTV system from scratch requires some basic knowledge of the method and experimental requirements. This document gives a brief description of the general design criteria of the PTV hardware (Chapter 2), a 'quick-start' guide for the PTV software *Streams* (Chapter 3) and details of my own PTV experiments (on the go in Chapters 2 and 3, or, in part, in Voermans et al. (2017)) In Chapter 4, I provide a brief summary and details on the use of the *Streams* files that should be provided with this document.

2 Experimental setup

It is difficult to go through the design criteria of the hardware components individually as each component impacts the design of the other elements. Nevertheless, in this Chapter it is attempted to describe the properties and design criteria of the tracer particles, the laser (as a light source) and the camera individually.

2.1 Tracer particles

Users should decide on the correct material and size of the tracer particles to ensure that the particles act as passive tracers and that the particles are large enough to be visible on the recorded images.

Tracer particles used should be acting as passive tracers in the sense that they follow the motions of the flow instantaneously. Though this criterion is often satisfied in PIV and PTV experiments by using very small particles with a density about equal to that of the fluid, it is worthwhile quickly calculating the Stokes number St. The Stokes number defines the ratio of the particle response time to changes in acceleration of the flow and should be much smaller than 1 (e.g. Xu and Bodenschatz (2008)):

$$St = \frac{T_p}{T_f} = \frac{1}{18}\frac{\rho_p}{\rho_f}\left(\frac{d_p}{L_f}\right)^2 \ll 1 \tag{1}$$

where T_p is the response time of the particle, T_f is the smallest time scale of the flow, ρ_p and ρ_f are the densities of the particle and fluid, respectively, d_p is the diameter of the particle and L_f is the smallest length scale of the flow. T_f and L_f relate to the smallest turbulent scales that need to be resolved. If the density difference between the particles and the fluid is negligible, a rough estimate is that the size of the tracer particles should be less than the smallest fluid motion of interest.

Another criterion on particle size is the number of pixels they occupy on the recorded images. The optimal recorded size of a tracer particle on images is about 2 px in diameter (Cowen and Monismith, 1997). That means that if the camera is moved away from the flume, the diameter of the tracer particles needs to be increased as well. Though this sounds obvious, it means that the size of the tracer particles is critically determined by the camera and the size of the measurement domain and can therefore be determined beforehand. Note that the recorded size of the particles are typically slightly larger then the actual size of the particles, which is caused by the glare that surrounds the tracer particles.

When selecting the material of the tracer particles, you might want to consider the shapes of the particles (round is preferable for the software and the response of the particles to changes in flow accelerations), colour (generally, the more reflection of light the better) and size distribution (narrow range of diameters is of course preferred). If uncertain about materials, check publications of recent experimental studies with similar configuration. While silver coated microspheres are popular, they are expensive and therefore only practical for small scale experiments. Though pliolite has been a popular choice throughout the years, the paint companies are currently using dissolved pliolite since and it is hard to get hands on the solid material (at the moment of writing). Do not forget that some material absorb small amounts of water, meaning that the dry and wet mass are not necessarily the same. Small buoyancy can, however, be influenced by increasing the fluid density with salt.

2.2 Laser

There are four important laser properties that need to be addressed when selecting a laser: wave-length, lens, sheet thickness and power.

2.2.1 Wave-length

The wave-length is not always a relevant specification in designing fluid experiments, but is relevant concerning laser safety (such as goggles) and when the wave-length is an important fluid property (such as for refractive-index matching and fluorescent dye).

2.2.2 Lens

Off the shelf PTV systems create a non-diverging and uniform intensity laser light sheet by using (moving) mirrors and lenses. Though these laser properties are considered 'ideal' in most experiments, they are not always relevant for every experimental setup. Comparable laser light sheets can be created with cheap laser pointers by including a lens in the laser housing. The main difference with the laser light sheets in off-the-shelf PTV systems is that light sheets from pointers are diverging.

Two lenses are able to create a laser light sheet of uniform intensity. *Fresnel lenses* are cheap but not continuous (note, this is not easy to see just by eye but very apparent on camera images). This means that tracer particles travelling through the light sheet are intermittently illuminated, which strongly reduces the success rate of the particle tracking software and hence the accuracy of the PTV results. A slightly more expensive but continuous line can be created by a *Powell lens* and is therefore recommended. Be aware that any cheap laser that creates a line or sheet uses a Fresnel lens (e.g. Google for 'laser line pointer'), so make sure you are specifying your needs to the supplier/manufacturer.

The downside of including these lenses within the laser housing is that diverging light sheets have a reduced intensity with increasing distance from the laser. The angle of the diverging laser sheet is also known as the *fan-angle*. The width of the sheet is therefore narrow close to the laser and wider further away. If the intensity is insufficient far away from the laser, either the intensity needs to be increased or the fan-angle to be reduced.

2.2.3 Line thickness

Ideally, the laser light sheet is as thin as possible to avoid any contamination of lateral flow variations in the recorded images. However, the finite site of tracer particles and the relatively low tracer particles density in the flow requires the light sheet to be thicker. It is difficult to assess how thick the light sheet should be a priori and the easiest way to get an approximation is therefore by looking at comparable studies using PTV and PIV. As a guide, the light sheet should be $O(\text{mm})$. In my experiments, I used 1 mm thick laser sheet for tracer particles with a diameter of about 35 μm and 3 mm thick for 95 μm diameter tracer particles.

2.2.4 Power

The power is probably the most difficult parameter to decide on when specifying the laser as its choice depends on many parameters. While a high intensity laser leads to higher laser-safety requirements, it does greatly improve the quality of the recorded images and hence the PTV-results. In the end, the intensity determines the brightness of the tracer particles on the images and a minimum level of intensity is necessary for the software to identify the particles. Note that the actual local intensity depends on the distance to the laser, the fan-angle and the thickness of the light sheet. There is, however, an upper limit as well as you don't want to work with high power lasers in simple experiments. For instance, you can google a bit around to see what high power lasers can do (apparently a 500mW laser pointer can melt plastics and set paper on fire).

2.2.5 My lasers used

The lasers used in my experiments are 2 identical lasers (manually aligned by myself) of 50mW, 650nm, 60° fan-angle and adjustable thickness of the light sheet (1 and 3 mm used). They were custom made by www.worldhawk.com for 37 USD per piece and were of excellent quality. The only thing I would have changed about this purchase from hindsight would be a higher intensity intensity (probably to 100mW) as it would have improved the quality of the images even further and would not have required any additional laser safety measures (would have been 45 USD a piece). Considering the low cost of the laser, I would recommend not to purchase an adjustable light sheet thickness but buy two or three lasers with different but fixed thickness, as it is a very time consuming task to change the thickness to a specific width when not being allowed to see the light sheet just by the eye.

2.3 Camera

Choice of camera is dependent on the hydrodynamic conditions and scale of the measurement area. High flow velocities require a higher frame rate as the travelled distance of a particle

Camera	Prosilica GC2450	Prosilica GT2050	GX2300
Resolution	2448 × 2050	2048 × 2048	2336 × 1752
Sensor	Sony ICX625	CMOSIS CMV400	Kodak
Sensor type	CCD	CMOS	CCD
Max fps full resolution	15 fps	28.6 fps	30 fps
Max fps	30 fps	70-80 fps (for 800 px)	50-60 fps (for 800 px)
Electronic shutter	25us - 60s	1us - 126s	10us - 60s
Sensing area	8.4 × 7.0 mm	11.6 × 11.6 mm	12.8 × 9.6 mm
Pixel depth	12 bit	8/12 bit	14 bit
Pixel size	3.45 × 3.45 μm	5.5 × 5.5 μm	5.5 × 5.5 μm
Price (estimate 2015)	3450 AUD	5000 AUD	6000 AUD

Table 1: Specifications of three cameras. The Prosilica GC2450 used in my experiments.

between two consecutive frames becomes larger and hence more difficult to match with *Streams*. Increased flow velocity also requires a lower camera exposure time, otherwise particles will appear as 'lines' on the images. The downside of reduced exposure time is that the light received by the camera sensor reduces as well, making it harder to identify the particles on the images. Therefore, there is a broad range of camera specifications that needs to be considered. As an example, the specifications of three cameras (in my experiments I used the Prosilica GC2450) are presented in Table 1. The other two were identified as alternatives to the GC2450 for my experiments back in 2015. The resolution is of primary importance as it links the measurement domain to the correct tracer particle size. The resolution (number of rows) can be reduced to increase the frame rate (*fps*). Sensing area is an important property, and generally the bigger the better, just as the pixel size, as larger areas on the sensors can be exposed to more light. The main reason I considered a new camera is the limited frame rate of the GC2450. For instance, the maximum flow velocity in my experiments was 16 cm/s and the minimum window 9 cm. Considering a frame rate of 23 fps, that means that a particle travelling through the measurement domain is captured 12-13 times. Depending on the data requirements, this can be very limited. As many publications mention the details of the camera used for PTV (particularly the fps and camera resolution), it is easy to get an indication of the requirements for your own experiments.

I used *Streampix* to record the images. It requires license to run. In the software settings you can change the properties of the camera, such as frame rate, resolution (note if you decrease the number of rows of the images, you can often increase the frame rate) and exposure time. Make sure you write down all these settings when performing the experiments, as you might need them for repeat experiments. Sometimes *Streampix* gives a warning that an image is missed. This is probably because the memory is full and the speed of writing the images onto the disk is too slow. In that case it is worthwhile to use a SSD for the recording of the images.

2.4 Few notes on taking the images

It is of critical important to get the highest quality of images possible. Ideally, the captured particles are round and have a strong contrast to the background. Of course, this is not always possible due to limitations of the experimental setup and equipment. When particles appear slightly elongated in the image, the PTV software will still be able to recognize them, but potentially as multiple particles. This should therefore be avoided as much as possible.

If particles are barely visible on your screen, it doesn't mean you cannot use the images. Sometimes the brightness of the computer screen or your eyes make it difficult to distinct between background and particles. I had to increase the contrast of all my images (easiest to do with *Matlab*). Objects that reflect light continuously can be removed in *Matlab* as well, by averaging all the images and then remove the averaged image from all the images. This can be done in either *Matlab* or *Streams*.

In the end, the quality of the data is determined by the quality of your images. If you can follow the flow patterns or tracer particles while scrolling through the images, there is no reason for the software not to be able to match the particles. If too many particles leave the laser sheet laterally, or the frame rate of the camera is too small such that particles move long distances between two consecutive images, you probably have difficulties in tracking them down yourself. If you are unsure about the quality of the images, take a batch of the images and run it all the way through *Streams* to make sure the velocity fields are of sufficient quality!

To give an example of a 'good' image, see Figure 1. In this image the density of particles is slightly lower than the recommended density by Cowen and Monismith (1997), nevertheless, PTV is much more forgiving than PIV on particle density (though this might impact the resolution of your results). The diameter of the particles in the original image are 2 − 3 pixels, but increases when both contrast and brightness of the images are changed. Figure 1a shows the original image. Note that the frame rate at which this image is obtained was low, so the actual intensity of the particles is already quite high. Examples of higher image intensities of the tracer particles are shown in the other sub-images in the sequence. Running these images through *Streams* suggests that (a), (b) and (c) are all of sufficient quality and it is this quality that the reader should aim for. Though many of the particles are still identified in (d), it becomes more difficult to correctly identify all particles. The reason why (d) turns out problematic is that the relative intensity of the particles (the intensity of the particles divided by the intensity of the background) actually reduces compared to the other images.

I recommend that after each experiment (or before) you measure the measurement domain with a ruler. That is, you put a ruler inside the flume in the laser light sheet, and take a picture with the camera. Then you know exactly what the size is of the image and hence a pixel. If you change the position of the camera or the laser, you have to measure this again. This way you can always trace back what the exact pixel size is and then you are not depending on less accurate estimates based on measures such as the water depth, dowel size, particle size etc.

3 Streams

Using the PTV software *Streams* correctly is a very laborious task in the sequence of experiments to usable data. While I try to be as elaborate as possible in the steps that need to be followed to go from images to velocity data, it should be clear that the software *Streams* comes with a very comprehensive set of manuals and a detailed description of all the theory and options embedded in the software. The purpose of Chapter 3 is merely to give new PTV users a push in the right direction and share my experiences. As every flow typology, configuration and even different flow velocities can lead to significant changes in the *Streams* calculation schemes, the user should, at some point during their studies, read significant parts of the manual to ensure that the experimental results are of the highest

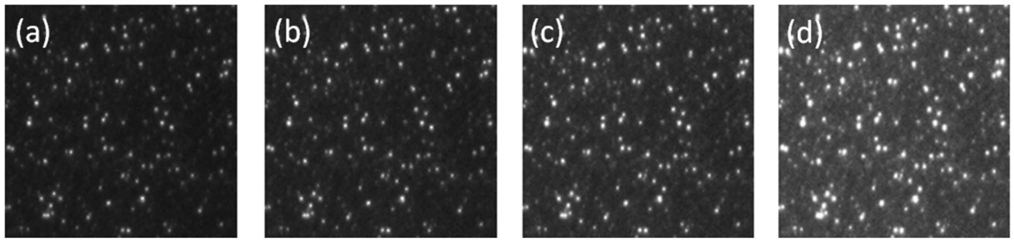

Figure 1: Examples image enhancement of captured particles. (a) is the original image. Contrast is slightly enhanced for (b), further enhanced for (c), and more extremely for (d). Note that many different techniques are available to change the brightness and contrast of an image. Particles have an average diameter of 2 – 3 pixels in the original image but this increases when brightness and contrast are enhanced.

quality possible.

3.1 Quick start (detailed)

This is a step-by-step guide to follow when you have obtained a set of quality images.

- Start with importing the images into *Streams*, click on 'Image', then 'create image sequence'.
  - Define time step, and pixels size (mm/px) (example values are given in Figure 4, but note that these should be measured in each experiment).
  - Go to the tab 'Image files' to add image files (sometimes it is easier to start with a small batch of 100 images or so). Tip: in my experiments I split each recording into multiple batches, otherwise the file size would become too large at a later stage, causing issues in saving these files or the software will simply run out of memory during calculations. For my experiments, each batch had a size of about 3000 images.
  - When pressing OK, 'Image sequence' appears in window on the left.
- **Image Record**: Double click on 'Image Sequence' (or alternatively, the name you gave it) opens the images/pictures on screen. Note that you can still change the details of time step and pixel size at this stage (right-click on the Image sequence and choose 'open details view'), but not later on.
  - Right click on 'Image Sequence' and choose 'Open process view'.
  - By clicking 'new', you can select one of the options on the left side. For instance you can create long exposure images, average images and subtract images from one another. Next step is to 'Identify particles'.
  - Select 'Identify particles'. Various particle identification (PID) methods are available. 'Dual threshold' is the preferred PID (Dual Gaussian is an option but is more time consuming and doesn't always work better). Single threshold is an option but as the name indicates, uses only one threshold to identify particles. This option seems to work best if your particles appear elongated (as in lines) on the images.
    - Choose 'Dual threshold' and click on tab 'Algorithm'. Choose here the peak threshold intensity (which is the minimum intensity of a pixel or group of pixels necessary to be identified as a particle).
    - The size of the particle is then determined by the Edge threshold factor (i.e. a fraction of the peak threshold intensity). Note that an Edge threshold of '1'

    is effectively the same scheme as the 'Single threshold' identification scheme, as particles are identified by patches of pixels where each pixel has an intensity larger than the threshold defined.
    - Minimum and maximum diameter can then be used to remove identified particles that are too large or too small to be real particles.
    - In the tab 'Frames', you can determine which images of the image/picture sequence should be included in the particle identification (e.g. only the first 100 or something). Default is all images.
    - In the 'Regions' tab you can include or exclude region from the images, that is, these regions define the area in which the PID schemes are applied exclusively, or ignored.
      - To create a region, go back to the image/picture itself. Press 'shift' and click and drag the mouse over the screen and you will see a rectangle appearing. When you release the mouse, the edges are either red or blue, meaning an internal or external shape. In the tab 'Region' you can choose different shapes, internal or external, and invert from internal to external or vice versa. Double click in the region to select a region if necessary. You can use Alt+arrow buttons to change the size of the region, and Ctrl+arrow buttons to move it. Save the final regions. You can then open the saved regions in the tab 'Regions'.
    - When finished with the PID, click 'ok'.
  - The 'Dual Threshold PID' appears now in the process window, and can be executed with right click 'Execute'.
    - Tip: before executing the Dual Threshold PID, there is an option to see an example of the identified particles in the image/picture directly. Open the 'Image Sequence' with a 'double-click'. In the options directly above the image, there is a dropmenu 'PIDs'. By clicking on 'Dual threshold PID' (or any other PID you created) the window of the PID opens and you should see simultaneously all particles identified in red in the image on the background. Without closing this window, you can alter the threshold values or particles sizes under the tab 'Algorithm' again, and see directly the impact on the identified particles in the image/picture by clicking on 'Update'.
    - The time of execution (of the PID) depends on the number of particles, number of images and size of the images.
    - Tip: As the execution of the PID can take a few hours, my computer went into sleeping mode (I could not turn it off due to limited administration rights on my computer). You can download a 'mouse jiggler' (free software) to overcome this.
    - When the PID is finished, a 'Particle Record' appears in the left screen.
    - Tip: It is worthwhile to safe the Image Sequence by right click 'Save'. Then also the PID is saved within this file. As you can 'open' a saved Image Sequence, you can then always get back to your defined pixels size and time step properties as well as the PID used.
    - Note that if you execute the PID again (even without changes), a second 'Particle Record' will appear with the same name. So be sure you know which 'Particle Record' belongs to which PID. Of course, you can change the name of every 'Particle Record' that appears.
- **Particle Record**: The Particle Record contains all the information of particle coordinates and particle matches (right click, 'Open match summary view').

- By double clicking on the ‘Particle Record’, a view-window opens in which you can see all the identified particles. You can play the sequence (see play button above the image) to see the identified particles move.
- The next step is to match the identified particles between consecutive frames. Right click on ‘Particle record’ to ‘Open process view’.
- Click on ‘new’. Start with ‘PTV analysis pipeline’ from the ‘primary processes’ section. Choose a name, I use here ‘PTV1’. After clicking on OK, ‘PTV1’ appears in the process view.
- Open ‘PTV1’ with double click, which opens the pipeline. All the calculation schemes made by you will appear here, and the properties of the calculation schemes are described below.
- When clicking on ‘New’, a new calculation scheme can be added. First you have to choose which of the particles you want to consider for matching:
    - ‘Global optimisation’ takes into account all particles that are identified, including the particles that are already matched. Hence, when using a ‘global optimisation scheme’ after a ‘global optimisation scheme’ you will lose the matched particles of the first scheme.
    - ‘Residual optimisation’ includes all particles that have not been matched yet and excludes particles that have been matched.
    - ‘Clean up’ removes identified matches that are not following the calculation schemes provided by you.
    - Note, it is most common to use a ‘global optimisation’ first, then ‘residual’ and then a clean up. The fewer particles available for matching, the shorter the calculation/execution time is. In my case it was not uncommon for a ‘global optimisation’ scheme to take a few hours of time (again, don’t forget the ’Mouse Jiggle’ if you cannot turn off the screensaver).
  - Start with ‘Global’. Go to ‘Costings’ and choose one of the build-in particle matching calculation schemes. It takes some time to figure out what the different schemes do, how they work, and when they are working best. The manual of the software is pretty helpful and detailed here. ‘Distance’ typically works well when cameras have a very high frame rate, as particles do not move that much between two images. ‘Correlation’ works pretty well in general and correlates the composition of particles between two images. ‘Pseudo-correlation’ works almost similar but is a more time effective scheme than ‘correlation’. Sometimes a combination works really well, such as ‘Pseudo-correlation’ and ‘Intensity’. At this stage, you cannot use the schemes based on velocity (Matching-based costings) as these matching schemes require knowledge of the velocity field which is not known a priori..
  - As an example, just pick ‘Pseudo-correlation’. Click on ‘Settings’. These options are about the details of this calculation scheme only. For ‘Pseudo-correlation’, the options of window width and height represent the region-size in which the composition of particles is compared to the composition of particles in the next image (it is difficult to give example values here as they are fully determined by the experimental setup, however, in Figures 4-8 some values are provided for the example images). If the window is too large, the particle composition in the window changes too much due to large scale turbulence, which affects the correlation negatively; if it is too small, there are too few particles to compare the compositions between the images. When you choose the numbers, just make

sure there are enough particles in such a window. It will probably take a few trials to have proper matches. It is often easy to recognize wrong matches, as they present themselves as non-smooth trajectories. To see if the particles are properly matched, it is worthwhile to quickly calculate the velocity field based on the initial matches. Based on the bulk velocity of the flow (determined either by the external parameters such as the flow discharge, or visually by tracking down a few particles) you can compare the estimate to the calculated velocity.

- Now click on the tab 'Optimisation'. Here you define the properties of the calculation in general. For instance, the search window, which determines the region in which the software looks for the particle to match. If you know that the mean velocity is say 7 cm/s, you can calculate based on the frame rate of the recording, what distance away the particle should be in the next frame, so don't make it too small (because turbulence can change the instantaneous velocity) and not too large (will be very time consuming and promotes incorrect matches). The window that defines the matching area is defined by 'Left', 'Bottom','Width' and 'Height', note that left and bottom are in reference to the particle that is going to be matched. So if there is large mean flow, it is unlikely that the particle moves against the flow direction, which means that the left boundary of the matching area does not necessarily need to be negative (when flow is from left to right).
  - E.g. in a region where the mean streamwise velocity is 20 – 50 mm/s (50 mm/s far away from the wall, 20 mm/s closer to the wall.) with a frame rate of 23 fps, particles are expected to move 1 – 2 mm each between two frames. I used in this case for the 'Pseudo-correlation' scheme a width of 8 mm and height of 4 mm (of course, this will depend on your particle density etc.). In the 'Optimisation' tab I defined the area in which *Streams* looks for the matching particle as: Left= 0 mm (particles are not expected against the direction of the flow), bottom= -1.0 mm, width= 5 mm and height= 2 mm. Note that these values are quite generous as the expected movement between two frames is just 1-2 mm. I could have easily lowered the width of the window to say 3 mm, and possibly the left corner to +0.5 mm. Also, in a nice turbulent boundary layer, vertical velocities are typically very small, making the height of the window of 2 mm relatively big, and 1 mm is in that case acceptable as well, maybe even 0.5 mm.
- There are also some different iteration routes to match particles, such as forward (i.e., start with image 1 to match to 2), backwards and others. Sometime one works better than the other so just try it out. Note that it can become difficult to match particles in regions where these particles enter the frame (for example, see the relatively large black area in the upper left corner of Figure 8). This can be optimised by repeating the calculation schemes (of course, use the residual optimisation otherwise you lose the already matched particles) and use the backward iteration direction.
- I have found the 'Regions' tab very helpful here. When flow properties are different in certain areas of the image (e.g. close to a boundary or recirculation zones), a different costing scheme works better. By using regions you can specify this separately. In my case, I used three different regions, separating the image in three layers. The top one had a mean streamwise velocity of 20-50 mm/s, the middle region 5-20 mm/s and the bottom one 0-5 mm/s. The lower two regions had relatively large vertical velocity components as well.

- Click Save and close, and close the PTV1 screen to get back to the process view. You can try the calculation scheme now by right click on PTV1 and 'execute'.
- When finished, you can double click on the 'Particle Record' in the left screen to open the image with identified particles. Make sure you activate the button 'Paths' above the image. If you move an image forward, all the colours of the trajectories might change, so sometimes it looks like new trajectories are shown but that's not the case. By pressing the play button, you can see the particles matches in time.
- You can also see how many of the identified particles are matched by your calculation scheme. Do this with a right click on the 'Particle record' and open 'open match summary view'. Of course, these numbers include both bad and good matches. However, I find it a useful option to see what percentages of particles is matched, i.e. whether it is worth doing a(nother) 'residual optimisation' scheme to try to increase the number of matches.
- Note that some subjectivity in PTV is still involved here, particularly during the design of the particle matching schemes where the user decides whether the matched particles are likely to be true matches.
- When you obtained an initial set of matched particles, you can finally use the velocity and acceleration based matching schemes (i.e. the lower part of the matching schemes you can choose from). Here, information of the particle velocity of previously matched particles is used. I think these velocity-based-schemes are the best way to match your particles. It uses the history of already matched particles to match new particles and seems therefore a robust way to identify the residual particles. For instance, if you have only one matched particle in the whole measurement domain, this matching scheme turns out to be quite effective already. That is, it is very likely that the particles most adjacent to this matched particle have a comparable velocity. This calculation scheme uses this velocity to estimate the velocity of neighbouring particles. If you would run this scheme a few times, it is likely you see the number of matched particles increase significantly.
- You will always find some particles that do not have a smooth trajectory (i.e. bad matches). It is these matches you don't want to have, but you also don't want to lose the properly matched particles. These particles often have a sharp change in direction (like an N-shape or something). It is relatively easy to remove these by adding a 'clean up optimisation'. These sharp turns in direction is basically a very large change in acceleration and/or directional velocity. A good first-try would be a clean-up with 'recent acceleration' and/or 'recent velocity'.
- Sometimes (if you are lucky) one 'global optimisation' is sufficient. This typically is the case when the flow structure is relatively simple or frame rate is very high (in other words, if it is really easy to spot matches with the naked eye, the software finds it easy to match too).
- Don't get too annoyed with too few matches, even after many different attempts. In Figure 2 below you can see the number of matches I get after a matching scheme combining: Distance, Intensity and Pseudo-correlation. If you zoom in enough on the image you can see the tiny white dots (the identified particles). I performed a 'global optimisation' scheme based on the local velocity of these few matched particles. That means I throw away the few matches I already have, but still retain the local velocity information of those matches for my velocity-based matching scheme. In every turn a few more particles surrounding the matched

particles get matched, so you might want to repeat this scheme a few more times. After some clean-ups and residual optimisations, about 90% of the particles were matched (see the trajectories in Figure 3, top right corner). Note that 90% is reasonable, as there are always particles that move out of frame, are not properly captured by the PID or are just not matched.

  - In Figure 3 you can see the final matching scheme used. Even though it looks like a chaos of matching schemes, the six different schemes used (i.e. General1, General0, ... Expand-1), apply to specific regions in the image, and contain many residuals optimisations and clean-ups. As seen, the velocity based matching schemes are very helpful and can be repeated many times until no additional matches are found.
  - It pays off to pay a lot of time on the right schemes. At some point you get some intuitive feeling what works best in which case, and sometimes you can reuse schemes in different experiments. It is this process that consumes most of your time. Just make sure you are writing down (keep track of) what you are doing and what works.
  - Tip: when you have many Particle record processes in line, for example as in Figure 3 (General1, General0, ... Expand-1), you can run them all together by putting them in the pipeline (that is the empty white window below in the Particle Record process view. You can do that by selecting the scheme (e.g. General1) and then hit the button 'Add to pipeline'). Then 'Execute Pipeline'. Then you can take a coffee break. Even though you can technically add all schemes, from the particle identification up to the calculation of the velocity fields into one pipeline, I would not recommend this. If an error occurs somewhere in the pipeline (such as insufficient memory available, which happens every now and then), you will loose quite some time. Also, you won't be able to see or save any of the files in between, such as the Particle Record.
  - Tip: it is highly recommended to save the particle matching schemes (e.g. General 1, etc.).
  - Tip: when happy with the matched particles and schemes used, it is important to save the Particle Record for future reference! Then you can always load the Particle Record again to assess the data (for instance, to export the Lagrangian velocity data) or change the matching schemes and re-execute.

- **Velocity Record**: Now it is time to calculate the Eulerian velocity fields. Open a new process and click 'Create velocity field'.
  - Many options are present, but initial values are generally sufficient. Under the tab 'Grid', you can find the grid size of the velocity field. If you have many matched tracers particles in each cell of the grid, interpolation of the matches to the nodes occurs. Sometimes you want to change the interpolation scheme under the 'Interpolation' tab. It is good to consider the interpolation option 'Binned', recommended by Cowen and Monismith (1997). It puts a maximum to interpolation distance to the grid. So if you have a really fine grid, but only a few particles matched, you will end up with a velocity field with only a few nodes giving a velocity value, instead of interpolating the limited matches to all grid nodes. If you have high density of matched particles, there is no difference between the 'Binned' interpolation scheme and the more common 'Triangle Based'.
  - When satisfied with the options, click on execute and a new velocity record appears in the left screen.

  - Double click on the 'Velocity Record' to open, and you can quickly calculate the flow properties. You can create velocity profiles by averaging in space and time if you want. Note that plotting velocity fields or velocity profiles is an easy way to verify whether the matching schemes are reasonable or not. It also is a good way to see how much impact additional matched particles have on the actual velocity fields, that is, refine the particle matching schemes and re-calculate the velocity field and compare with other particle matching schemes.
  - I find it easiest to export the full velocity fields to *Matlab* and analyse the data there, though the visualisation options in *Streams* are useful to get a feeling of the initial results. To export the velocity fields, double click on "(u,v)", keep x, y and t as variable, and press calculate. Sometime this takes a few seconds to a minute depending on the size of the file, so don't think the software is frozen or not working. Right click on the field that appears to 'save text to file'. Click on 'Browse' and give the file a name. Default is csv, if you let it end with something else, like .dat, it becomes a .dat file.
  - Note that if the file is large, it can take up to a few minutes to save. If it is too large, *Streams* will run out of memory, and you have to save a fraction of the velocity fields. Reading a big file with *Matlab* is a pain but not impossible. Therefore, easiest way is to split long video records into batches as I mentioned previously. Make sure though that you are not skipping a time step when attaching the velocity data files back together again.
- To end this list, I would like to reiterate that all information about *Streams* can be found on http://www.civil.canterbury.ac.nz/streams.shtml, supplied by Dr. Roger Nokes, as many more options are available.

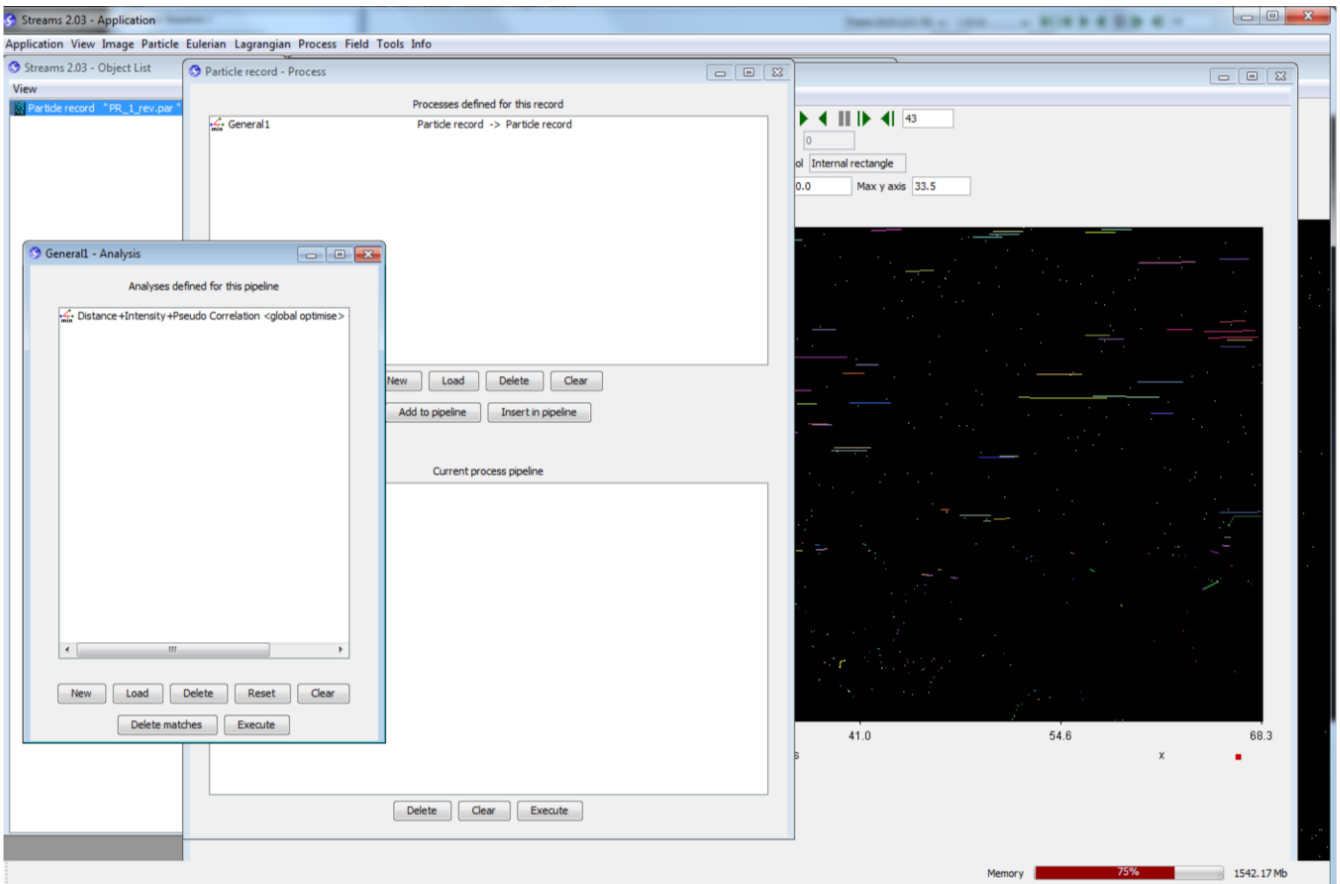

Figure 2: Your matching scheme is not necessarily useless when only a few matches are obtained. This is an example of the number of matches I obtained after running one scheme. You can always add more schemes to get more matches.

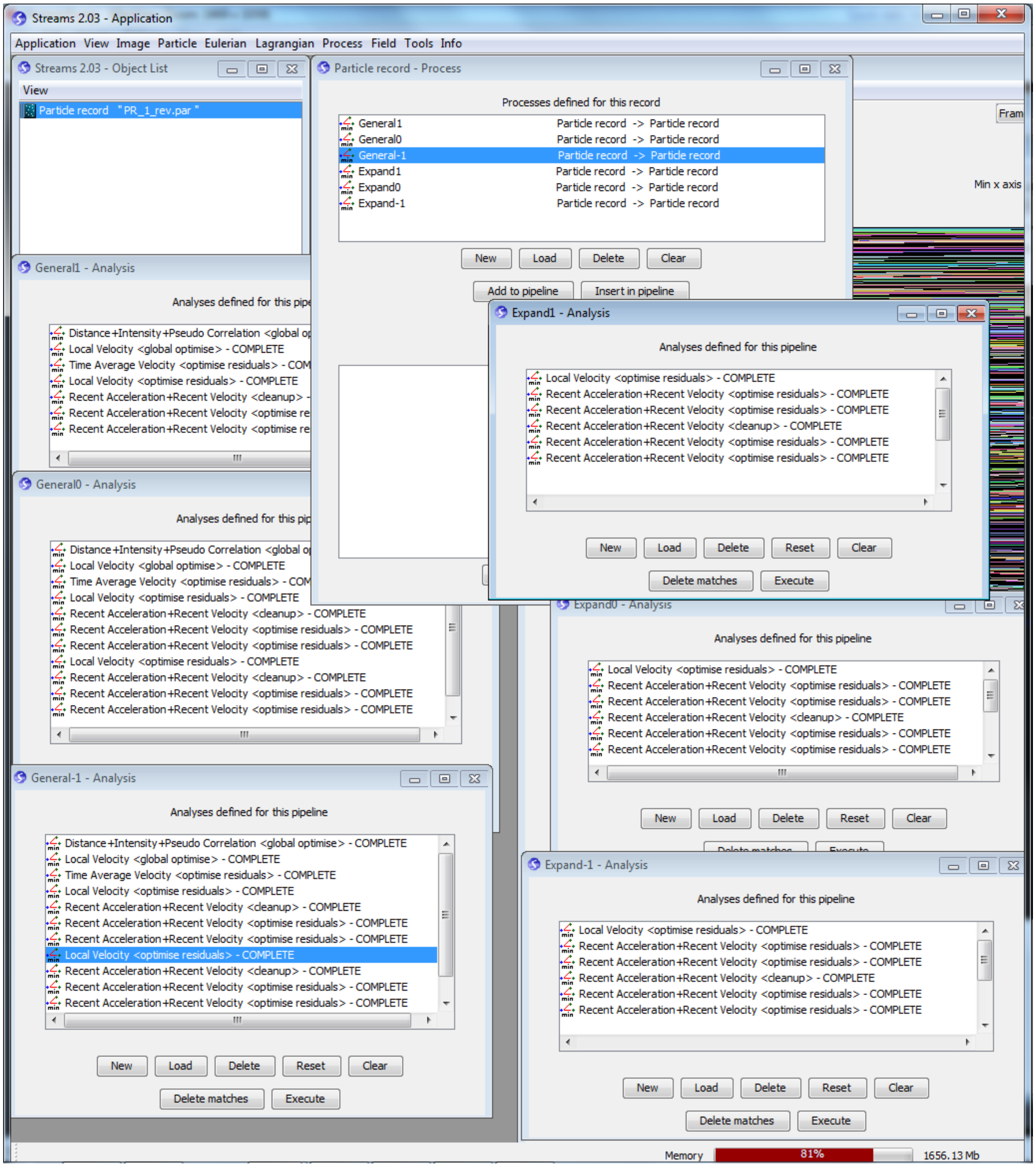

Figure 3: This is the final matching scheme used for an experimental case. Note that these subschemes are used over specific areas within the measurement domain.

4 Quick start (summary and example)

In Figure 4, a summary workflow is presented. On the right-hand-side of the workflow, instructions are provided on the use of the supplemented example files. The example files are 20 images on the flow over a backward-facing step, and additional files for *Streams*. Note that Figures 5-8 illustrate the impact of different 'particle tracking schemes', see the captions of these images for more detail of the schemes. In Figure 9, the mean streamwise velocity is given, based on 30 images.

(This document is written by Joey Voermans, 12-02-2018)

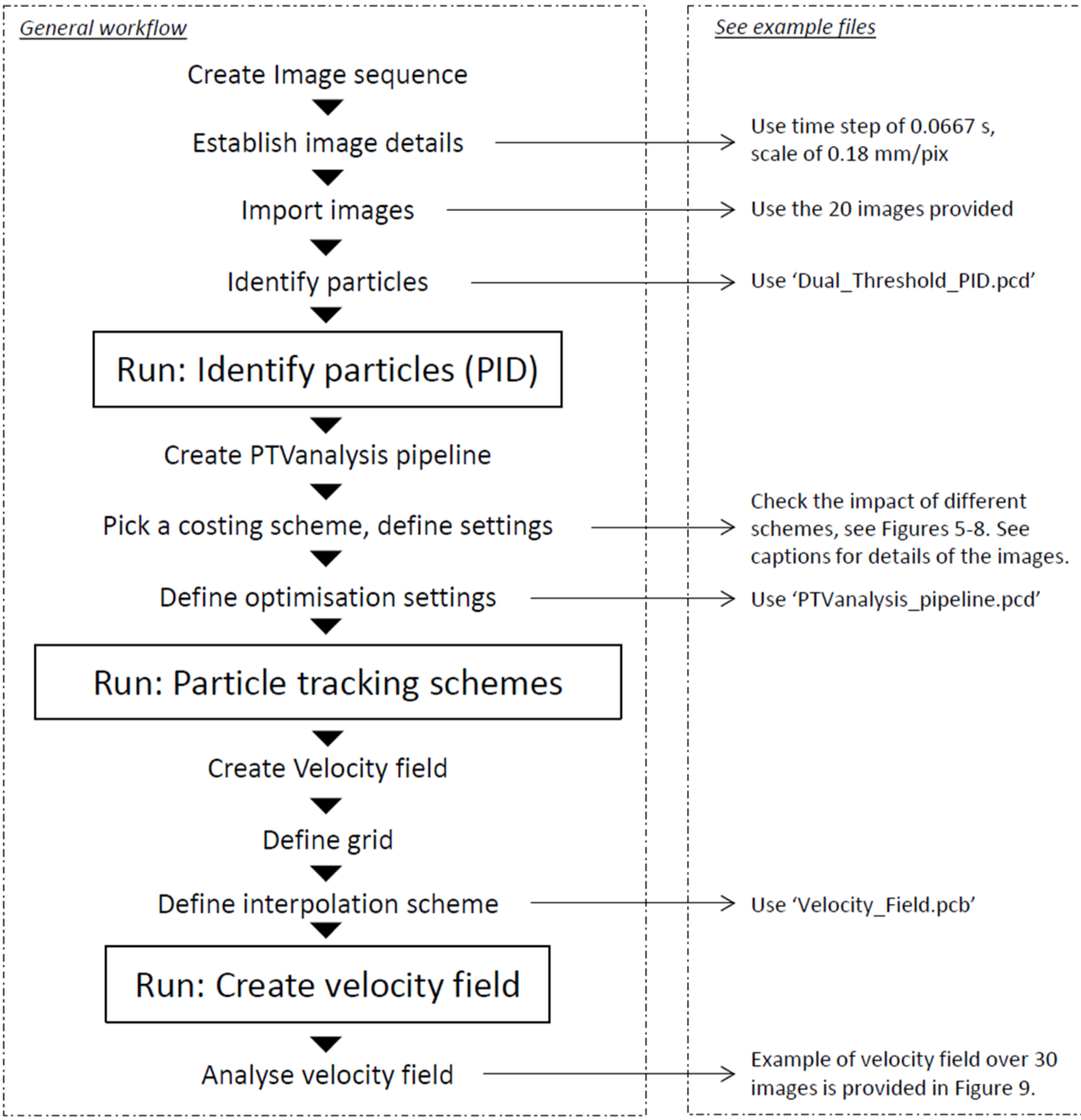

Figure 4: General workflow on the use of *Streams*. The right part of the workflow features instructions on the use of supplemented example files of PTV images of a backward facing step.

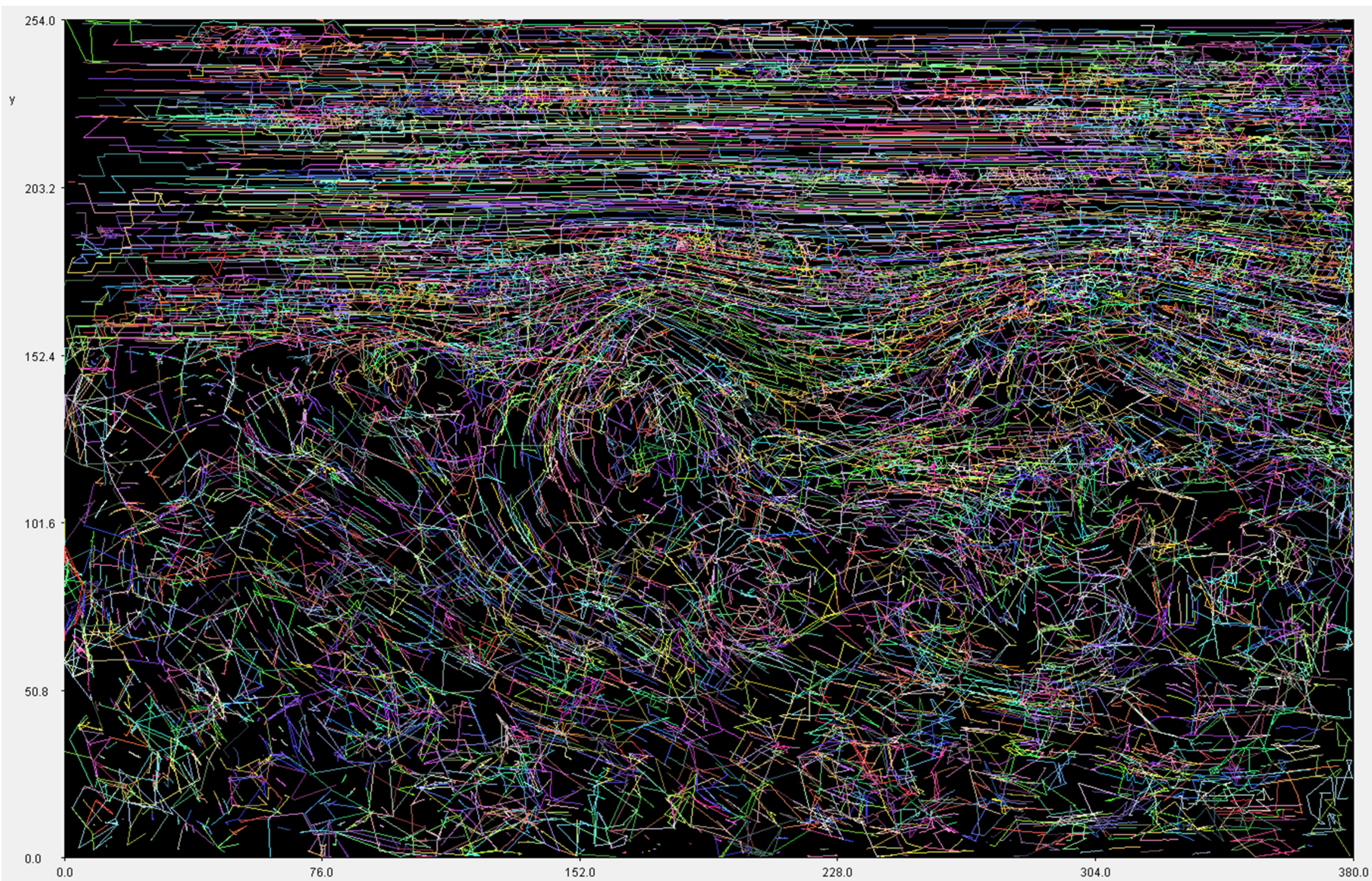

Figure 5: Identified matches between particles for the ‘particle tracking scheme’: Distance, with optimisation: Left= −20, Bottom= −15, Width= 40, Height= 30. Note that many particles are incorrectly matched.

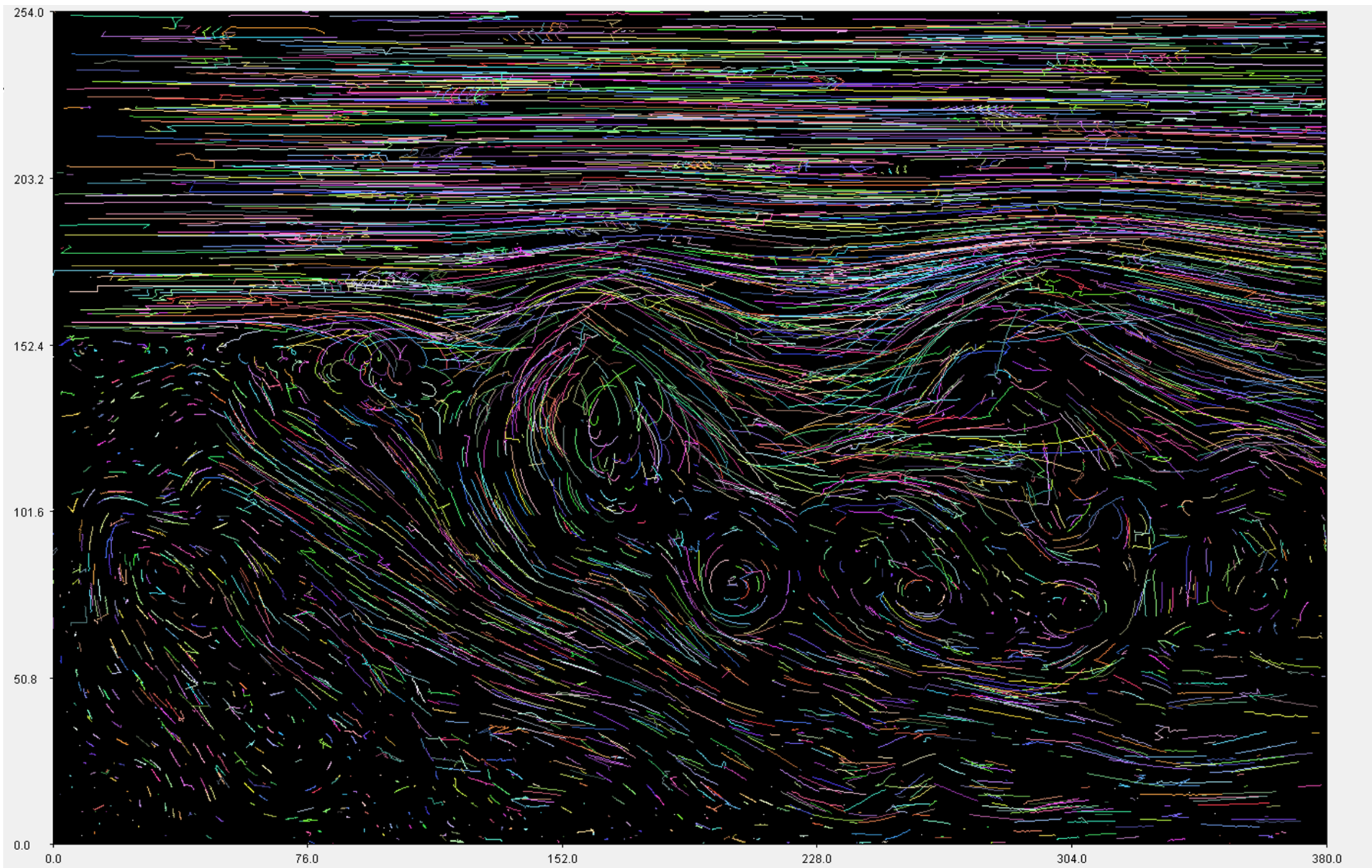

Figure 6: Identified matches between particles for the 'particle tracking scheme': Distance, with optimisation: Left= -5, Bottom= -2, Width= 10, Height= 4. Compared to Figure 5, the majority of particles are now matched correctly. In this scheme, the window in which the potential particle match is looked for is reduced in size. Based on visual identification, the maximum distance a particle travels in between two frames is about 3 mm. To ensure that variation to this distance is still captured, the window is set to a distance of 5 mm (i.e. Left= -5 and Width= 10). Though this scheme performs well (because the flow velocity is relatively low, or conversely, the frame rate is high), still some bad matches are observed, in particular in the top part of the figure.

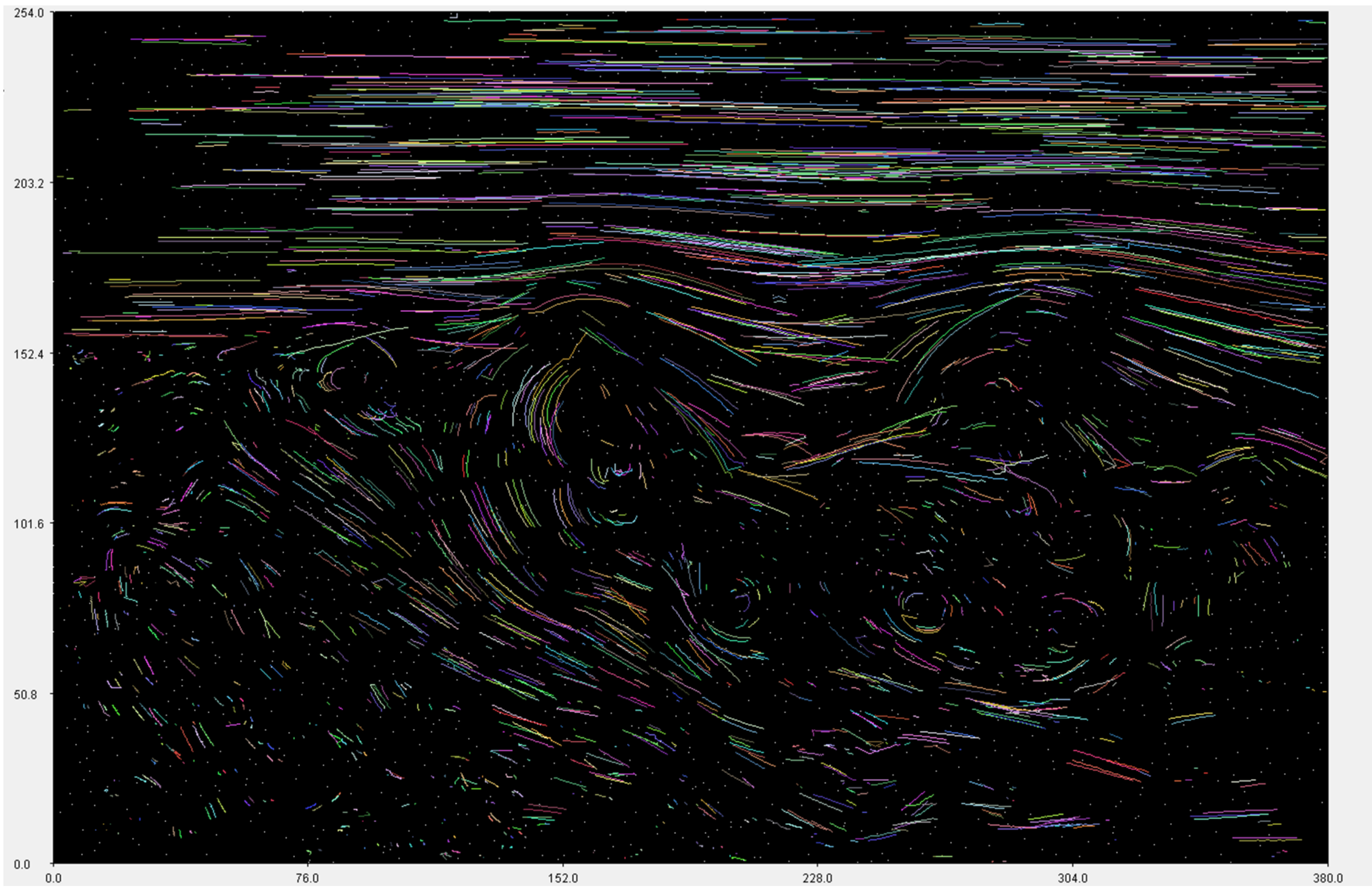

Figure 7: Identified matches between particles for the 'particle tracking scheme': Pseudo-correlation, with specific details: Window width 8 mm and Window height 3 mm. Optimisation: Left= −5, Bottom= −2, Width= 10, Height= 4. Compared to Figure 6, a larger proportion of the matched particles seem correct (i.e. clear flow patterns can be observed, consistent with that expected from flows over a backward facing step), but only a limited number of particles are matched (i.e. only 54% of the particles are matched). This is due to the relatively small window over which the pseudo correlation window acts (8 by 3 mm). If the particle density is too low, not many particles are available to establish the correlation between two sub-windows. This causes patches of matched particles, most obvious in the lower left corner.

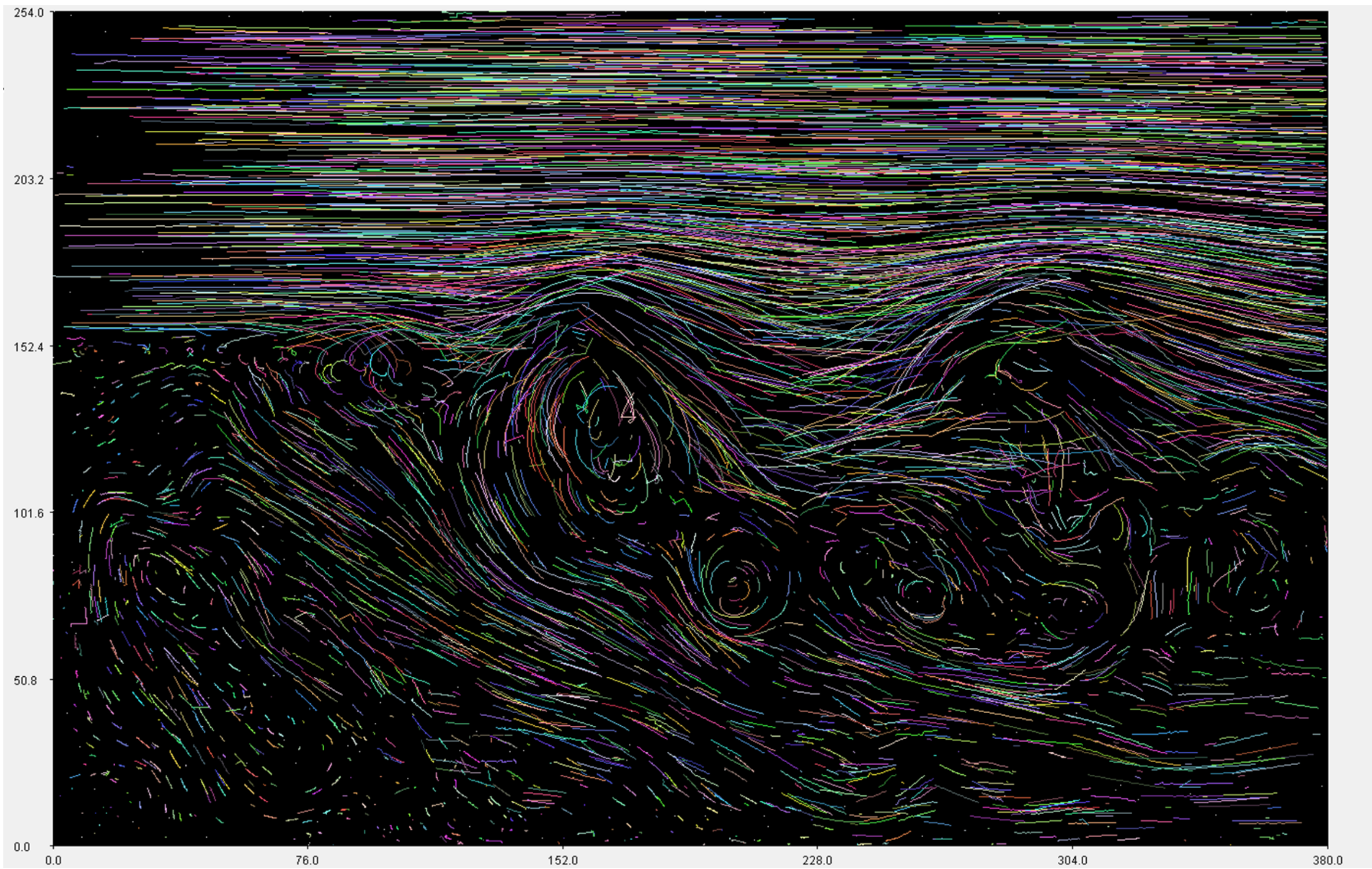

Figure 8: Identified matches between particles for the 'particle tracking scheme': Pseudo-correlation, with specific details: Window width 20 mm and Window height 8 mm. Optimisation: Left= −5, Bottom= −2, Width= 10, Height= 4. Compared to Figure 7 the pseudo-correlation window is increased to ensure enough particles are present in the sub-windows. Visually all particles are matched properly (only incidental incorrect matches can be observed). Here 93% of the particles are matched. Note that the flow structure is identical as Figure 6. Hence, this is a good scheme for this experimental case.

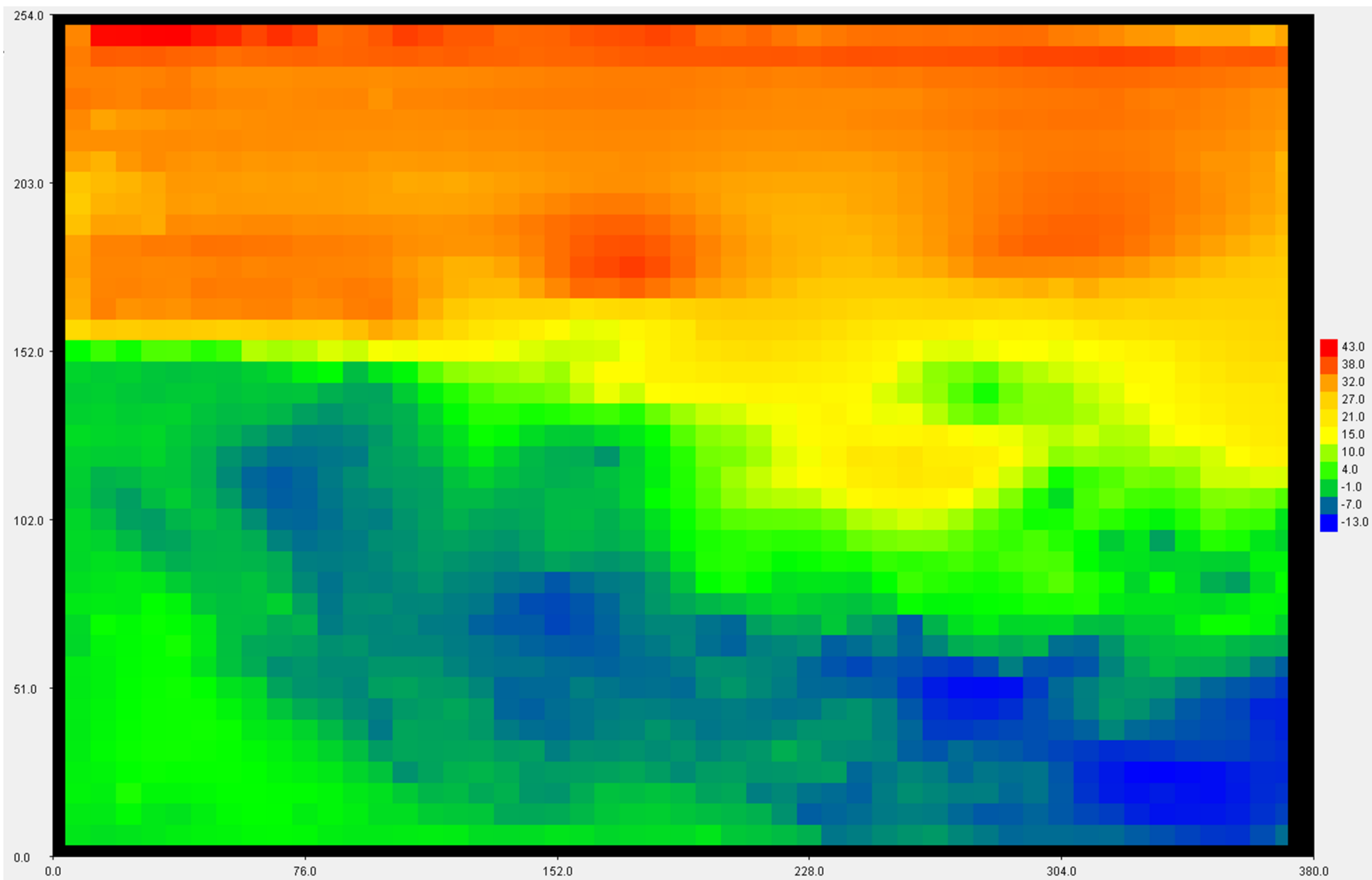

Figure 9: Velocity field calculated based on the particle tracking scheme used in Figure 8. Note that the maximum mean streamwise velocity is about 43 mm/s (2.9 mm per frame). This is close to the travelled distance of 3 mm in between two images (based 15 frames per second) as identified for Figure 6.

References

Cowen, E. A. and Monismith, S. G. (1997), 'A hybrid digital particle tracking velocimetry technique', Exp. Fluids **22**(3), 199–211.

Voermans, J., Ghisalberti, M. and Ivey, G. (2017), 'The variation of flow and turbulence across the sediment–water interface', J. Fluid Mech. **824**, 413–437.

Xu, H. and Bodenschatz, E. (2008), 'Motion of inertial particles with size larger than kolmogorov scale in turbulent flows', Physica D: Nonlinear Phenomena **237**(14), 2095–2100.