Tomo-PTV with sparse tomographic reconstruction and optical flow

P. Cornic, F. Champagnat, A. Plyer, B. Leclaire, A. Cheminet, G. Le Besnerais

Tomo-PTV with sparse tomographic reconstruction and optical flow

Philippe Cornic[1,*], Fredéric Champagnat[1], Aurelien Plyer[1], Benjamin Leclaire[2], Adam Cheminet[2], Guy Le Besnerais[1]

1: Department of Modeling and Information Processing, ONERA the French Aerospace Lab, France
2: Department of Fundamental and Experimental aerodynamics, ONERA the French Aerospace Lab, France
* Correspondent author: philippe.cornic@onera.fr

Abstract. The framework of this study is the conventional four cameras two time steps Tomo PIV. The rationale behind this paper, and its main contribution, is that using two ultra sparse tomographic reconstructions (one particle –one voxel: LocM-CoSaMP) at time step t and t+dt where the particles are only represented by the voxels they are embedded in, one can reliably link the particles in the two time steps and filter out most of the ghosts. It is acheived by driving the 3D particle matching by a measure of the 2D displacement on the images themselves (optical flow). To derive an accurate vector field, the positions of the matched particle are then refined by a global optimization to yield subvoxel location. The 3 steps of the proposed method are described. Synthetic experiments on a linear displacement and on a vortex ring are presented. The performances of Nearest Neighbors matching and optical flow matching are compared. The performances on the derived vector field are assed in terms root mean square error. We show on illustrations how well interpolating the vector field on a regular grid captures the structure of the flow.

1. Introduction-Motivation

This paper deals with 3D velocity measurement using a two-frames/four cameras PIV setting. Such a setting is actually typical of tomo-PIV (Elsinga 2006), but our 3D velocimetry method shares with 3D-PTV (Maas et al. 1993) the same philosophy of extracting 3D particles while working at much higher ppp than this latter. It has long been recognized that extracting 3D particles using early 3D-PTV (Maas 1993) techniques based on image detection and matching using epipolar geometry would break down at ppp larger than 1e-3 (Wieneke 2013), due to overlapping particle images and ambiguities in matching. This led Schröder et al (2009, 2011) to build 3D-PTV schemes based on 3D particles extracted using tomo-PIV 3D which is reported to operate efficiently in real experiments with ppp equals 0.05. Such an approach was later referred to as tomo-PTV by Scarano (2013), borrowing from Wieneke (2011).
Our method is closely related to the tomo-PTV paradigm, the vector field derivation relies on 3D particle matching in successive time-step, but in this paper we only deal with 2 time steps in view of forthcoming tracking.
Our method starts with an ultra sparse tomographic reconstruction that yields a one-particle-one-voxel detection where particles (and ghosts) are represented by the voxel they belong to. Such reconstructions are performed independently at time t and t+dt. The 3D particle matching is then driven by a consistency criterion built on 2D image motion in each camera. This process filters out most of the ghosts and yields reliable matchings. To derive an accurate particle velocity estimate, the voxel-scale 3D location of the matched particles is then refined by a global optimization to yield subvoxel location. Finally the sparse velocity map may be interpolated to yield a 3D velocity map.

1.1 Related work

The actual status of 3D PIV at high seeding relies on the reconstruction of 2-3 voxel large blobs using Tomo-PIV (Elsinga 2006, Scarano 2103). However, there is a growing interest in the 3D PIV community in pointwise particle representation instead of blobs. Cornic et al. (2013) proposed a sparse tomographic reconstruction based on a pursuit algorithm that yields a list of voxels that embed the particles, but without going so far as computing a vector field. Champagnat et al. (2013, 2014) used SMART but within a

paradigm favoring the reconstruction of particles instead of blobs. Vector fields are still estimated by 3D correlation.
Such pointwise representations are much more compact, encode particles with less memory and a better physical relevance: the tracers are often much smaller than the voxel scale. Furthermore, one might wonder whether correlating empty spaces is relevant. It imposes to take large interrogation windows to have a sufficient number of particles inside. Averaging over of the interrogation window acts as a low pass filter on velocity gradients and smooth out small structures. The greater the window, the greater the smoothing. In our procedure, dense 3D vector field is interpolated from individual particles velocities, this means that the amount of smoothing depends on the density of extracted particles.
Associated with time resolved (TR) PIV, the particle approach allows performing Lagrangian evaluations, hardly possible with Tomo-PIV. In order to do so, Schröder et al. (2009, 2011) detected subvoxel location of tracer particles in the time series of tomographically reconstructed volume. Instead of fitting 3D Gaussian on blobs to detect particles, we propose to use an ultra sparse tomographic reconstruction "one particle-one voxel" and to refine their location once they are matched.
Iterative Particle Reconstruction (IPR) (Wieneke 2011, 2013) starts with a standard 2D particle finding and 3D triangulation and then iteratively adds and removes particles and refines their 3D positions and intensities to match the projection of the particle distribution in the volume with the recorded images. However, this approach did not go as far as it might have gone, performing 3D correlation for the derivation of vector field.
Recently, Shake The Box (STB) (Shantz et al. 2013) predicts the positions of particles already tracked and refine their position with IPR scheme. STB is able to produce reliable long term tracks but requires in order working efficiently that a certain fraction of particles is already tracked. According to the authors themselves, the application of STB-method (or even IPR) to conventional two frames PIV is doubtful as no prediction is available and particle matching is difficult due to many ghost particles.
Eventually, the idea of using two time steps to get rid of ghost can also be found in (Novara et al., 2010), but in our approach, the consistency measure is built on the image space rather than on the 3D space as in (Novara et al., 2010).

1.2 Paper focus and organization

The paper is outlined as follows. Section 2 briefly introduces the reconstruction algorithm. Section 3 describes the main contribution of this paper, the optical flow driven 3D matching. Section 4 describes the synthetic setup. Section 5 is dedicated to the second contribution of this paper, the global subvoxel refinement.

2. One particle – one voxel reconstruction

The goal of the first step of the proposed procedure is to yield an approximate representation of the 3D distribution of particles at a given instant in the form of a voxel map with sparse entries. By sparse we mean that a particle should be represented by a non negative entry only in the nearest voxel node, in contrast to the usual blob representation obtained by tomo-PIV.
This "One particle – One voxel" goal is achieved by the LocM-CoSaMP introduced in (Cornic et al.,2013).

Similarly to tomoPIV, LocM-CoSaMP seeks a resolution of the tomographic linear system $\mathbf{Y} = \mathbf{W}.\mathbf{E}$ that relates the observations $\mathbf{Y}$ (pixels) to voxels intensity $\mathbf{E}$ through the weight matrix $\mathbf{W}$ (Elsinga, 2006)**.** But the One particle – One voxel goal is achieved in a very efficient manner by using jointly three salient features:

1. Building matrix W directly using the samples of the imaging Point Spread Function (PSF), this ensures that the image of a given particle can be represented by few coefficients (Champagnat et al. 2013) [MST],
2. Restricting voxel intensity E in system $\mathbf{Y} = \mathbf{W}.\mathbf{E}$ to local maxima of the MLOS map (Cornic et al. 2013)
3. Solving the system $\mathbf{Y} = \mathbf{W}.\mathbf{E}$ in the mean square sense allowing a maximum of S (the sparsity) non zero coefficients (i.e. using *sparsity* techniques)

The first feature is related to the Particle Volume Reconstruction (PVR) paradigm (Champagnat et al. 2013, 2014). PVR is based on the fact that the geometric image of tracers particles is usually smaller than the width of the Point Spread Function (PSF) of the imaging system, thus their intensity is spread around pixels nearby their projection coordinate in image plane according to the Point Spread Function (PSF) of the imaging system. The contributions of the different particles add. From these hypotheses PVR derives a discretization of the image intensity distribution in the form **Y** = **W**.**E** with **W** made of PSF samples, see (Champagnat et al. 2013). The choice of the voxel to pixel ratio (v/p) is an important issue with PVR. As shown in (Cornic et al. 2013), the reconstruction performances increase as v/p decreases and a good tradeoff between memory requirement and reconstruction quality is given by v/p = ½ .

The second feature is based on the observation/proof by Cornic et al. (2013) that particles are located in or close to the voxels which are local maxima of the MLOS function; see (Atkinson & Soria, 2009) for further details. Since within the PVR paradigm a single voxel can render a particle image, only the columns of the weight matrix **W** corresponding to local maxima of MLOS can be retained. Using This LocM step, less than 10% of the columns of **W** are retained (Fig. 1).

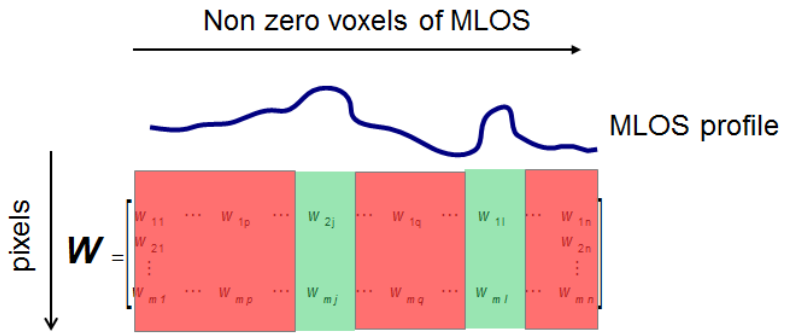

Fig. 1: Only the columns of **W** corresponding to local maxima (green) of the MLOS are retained in the system to be solved

The third feature is implemented through the use of pursuit algorithms dedicated to the search of sparse solution of linear systems. CoSaMP (Compressed Sampling Matching Pursuit) introduced in (Needell & Tropp, 2009) is a second generation pursuit algorithm. It iteratively solves the linear system **Y**=**W**.**E** in the least square sense under the constraint that the number of nonzero entries of E ($\|E\|_0$) is lower than an integer (the sparsity) set by the user.
Mathematically speaking, it solves:

$$\min_E \|\boldsymbol{Y} - \boldsymbol{W}.\boldsymbol{E}\|_2 \text{ under the constraint } \|E\|_0 \leq sparsity$$

Performances of LocM-CoSaMP have been evaluated in (Cornic et al. 2013) using an extensive synthetic data test. The study showed that detection performances outperform those of tomo-PIV in most configurations while the associated computational cost and memory requirement is much lower.
The reconstructions yielded by LocM-CoSaMP are not suited for correlation: the particles are represented by only one voxel (not blobs as in conventional tomography) and are located on a v/p=1/2 voxel grid. The displacement estimation has to be performed through the matching of particles at time t and t+dt. The matching amounts to determine which voxels at time t and t+dt represent the same particles.
The next section explains how one voxel particles extracted by LocM-CoSaMP are matched between adjacent time steps.

3. 3D matching

PTV techniques usually rely on accurate particle location prediction in order to get reliable temporal matches. This can be implemented through multiple frame techniques, eg. Minimum Acceleration over 3 frames, Minimum Change in Acceleration over 4 frames (Malik et al. 1993) (Ouellette et al. 2006). In the case of two frames PIV, such approaches do not apply, so the best prediction of one particle in the next

frame is its current position. In this context the simplest matching strategy is the Nearest Neighbor (NN) matching: match the particle detected at position ***x*** at time instant *t* with that particle which is the closest to ***x*** at instant t+dt. But as shown in the experiment (see § 4.2) it yields a lot of erroneous links.
Our strategy for bypassing the limitations of NN matching is to test a potential 3D match against 2D motion in each camera. Since the projection of a single particle can occur anywhere in image plane, the 2D motion information has to be *dense*. For this task we rely on FOLKI-PIV a fast, accurate dense optical flow algorithm (Champagnat et al. 2011), but the principle of the proposed matching method does not depend on this specific algorithm. For convenience, we denote hereafter the 2D motion in the images by optical flow without referring to any particular technic.

3.1 Basic principle of the optical flow driven 3D matching

The basic principle for driving the matching by the optical flow is well captured by the figure Fig. 2. It is illustrated on a four cameras setup which is the one used in the synthetic settup (see § 4.1). Each particle at time t is assigned a list of potential matches at time t+dt by taking all the particles within a certain distance, depending on the flow speed and time step dt. The list is then processed sequentially. The particle and a potential matching of the list are projected in the 4 images and the corresponding displacements are computed (cyan arrows). The consistency of the displacement field is then assessed against the measured optical flow in the x and y directions in the four images at the location of the projected particle indicated by a black cross Fig. 2 .
The consistency is assessed by a matching cost which is a function of the difference in each image between observed and predicted displacement.
Let $\left(x_j^t, y_j^t\right)$ be the coordinates of the projection in image j of the particle under consideration at time t, and let $\left(x_j^{t+dt}, y_j^{t+dt}\right)$ be the coordinates of the projection of a potential match at time t+dt. The consistency criterion reads:

$$\sum_j \left(max\left(\left|x_j^t - x_j^{t+dt} - Of_x\left(x_j^t, y_j^t\right)\right|, \left|y_j^t - y_j^{t+dt} - Of_y\left(x_j^t, y_j^t\right)\right|\right) < Threshold\right) \geq N_c \quad \text{(Equation 1)}$$

Where $Of_x \; and \; Of_y$ are the estimated x and y components of the optical flow. N_c is the minimal number of images where the computed displacement should meet the estimated flow: $2 \leq N_c \leq 4$
All the potential matches that do not meet Equation 1 are discarded and if any, the best one is validated as a match. Tie break for equation 1 uses the better consistency.
Depending on the complexity of the flow and on the thickness of the volume, the optical flow may be more or less well estimated depending on whether there are superimpositions in the correlation window of contrary movements originating from different depth in the volume. The cost may take into account this uncertainty in the optical flow estimation by relaxing the requirements to validate a match (see § 4.2).
The tolerance in term of discrepancy (Threshold) and the number of conditions to satisfy (N_c) are the two parameters one can play around with to take into account the error introduced by the localization on a voxel grid of the particles and the potentially inaccurately estimated optical flow.

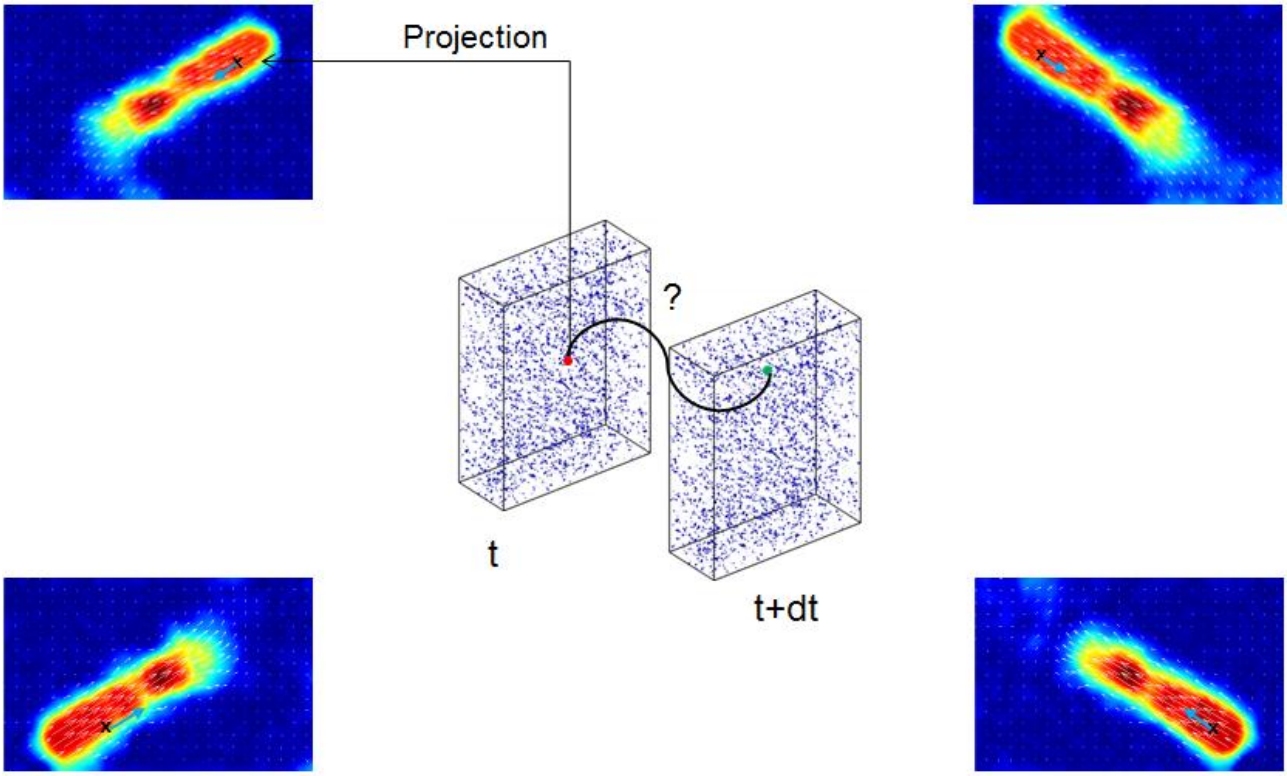

Fig. 2: Assessment of the consistency of the predicted displacement field associated with the potential matching (cyan arrow) with the measured optical flow (white arrow). The black cross indicates the projection of the particle at time t in the four images. The color codes the magnitude of the optical flow.

4. Performances for the 3D matching

LocM-CoSaMP yields a list of nonzero voxels occupied by a particle. A detection, i.e. a nonzero voxel, is a True Positive (TP) if it is in the neighbourhood of a true particle. Unless otherwise specified, the neighbourhood is here a 2x2x2 voxels whose centre chessboard distance from the particle is lower than one voxel with v/p=1 size. A detection is a False Positive (FP), ie. a ghost, if it is not in the neighbourhood of a true particle. A particle is recorded as False Negative (FN) if there is no detection in its neighbourhood.
A matching is referred to as a "good match" if its matches two corresponding TP detections (ie. two detections of the same particle at instant t and t+dt). As already done in (Cornic et al. 2013) to characterize the reconstruction, we define the Recall and the Precision measures for the matching. The Precision metric is the fraction of good matches among all detected matches. The Recall metric is defined as the fraction of detected good matches among all the possible good matches (detected or not).

$$Recall = \frac{\#detected\ good\ matches}{\#good\ matches} \quad Precision = \frac{\#detected\ good\ matches}{\#detected\ matches}$$

The best achievable performance is again given by Recall=1 (all the good matches are detected) and Precision=1 (all the detected matches are good matches).

4.1 Synthetic setup

A thorough description of our setup can be found in (Cornic et al. 2013, Champagnat et al. 2014). All our simulations involve four cameras, which are positioned on a single side of the laser sheet at the vertices $(\frac{\pm 1}{2}, \frac{\pm 1}{2}, \frac{1}{\sqrt{2}})$ of a square of 1 meter side. They are positioned at 1 meter from the centre of the reconstructed

volume located at (0,0,0) and point at it. The voxel-to-pixel ratio v/p = 1 leads to voxels of 0.1 mm side.
The tracers particles are uniformly distributed in the light sheet volume. It is important to notice that all the illuminated particles cannot be seen by all cameras, as in real dataset.
The images I_j (j=1..4) are synthesised according to:

$$I_j(\boldsymbol{x}) = \sum_p E_p h\left(\boldsymbol{x} - F_j(\boldsymbol{X_p})\right)$$

where $\boldsymbol{x} = (x, y)$ denotes any location in the image plane, h is the compactly supported PSF function, E_p and $\boldsymbol{X_p}$ are the intensity and 3D coordinates of particle p. $F_j(\boldsymbol{X_p})$ is the geometrical image of particle p. We assume that the PSF is a separable function. We use a single 1D PSF for both directions:

$$h(x) = (\Pi_\Delta * g_\sigma)(x) = \int_{-\infty}^{\infty} \Pi_\Delta(t) g_\sigma(x-t)dt = \frac{1}{2}\left(erf\left(\frac{x+\frac{1}{2}\Delta}{\sqrt{2}\sigma}\right) - \mathrm{erf}\left(\frac{x-\frac{1}{2}\Delta}{\sqrt{2}\sigma}\right)\right) \quad \text{(Equation 2)}$$

We take σ_{psf}=0.6. A gaussian noise with zero mean and standard deviation 2 is added to the images (with negative values thresholded to zero). Its amplitude is thus about 5% relative to the maximum particle intensity.

4.2 Synthetic experiment: Vortex ring

The displacement field considered is a vortex ring, similar to that considered by Elsinga et al. (2006), whose main section lies in the z = 0 plane. The radius of the ring is equal to 393 voxels and the maximum displacement is equal to 2.94 voxels (in v/p = 1 unit).
In this case, the optical flow – shown Fig. 3 for one camera and ppp=0.057 – is not easy to estimate because there are antagonist movements over the depth. This results in complex displacement in the image that may not be well captured with the window used to estimate the optical flow. In this case, the consistency of the displacement of a potential match is assessed against at least two optical flows out of four (N_c=2).

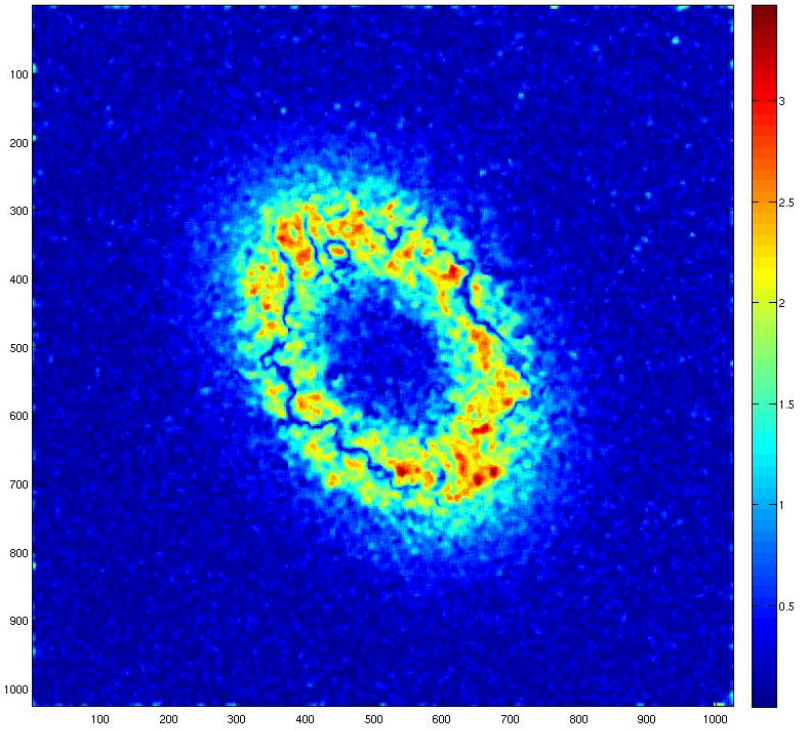

Fig. 3: Vortex ring experiment ppp=0.057. One out of four optical flows. The color codes the magnitude of the displacement in pixels.

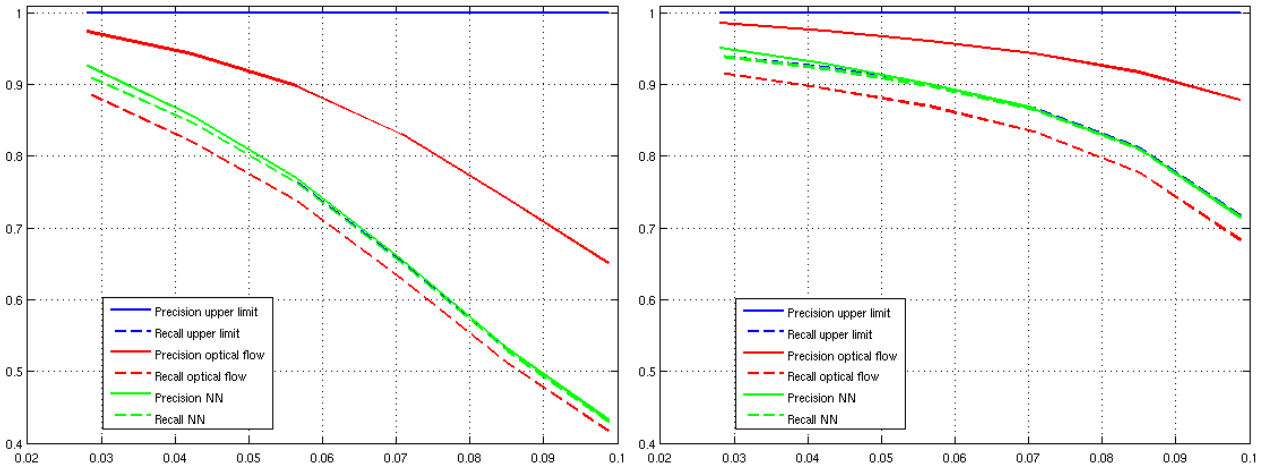

Fig. 4: Matching performance for vortex ring displacement field. On the left hand side v/p=1. On the right hand side v/p=1/2. On both sides, upper limit Recall and NN Recall are superimposed.

Fig. 4 shows the matching performances as a function of the ppp. On the left for v/p=1 reconstruction, on the right for v/p=1/2. The sparsity parameter is set to the exact number of particles in the volume. The solid and dashed blue curves give the best achievable Precision and Recall (upper limit) with the two computed reconstructions. It can be seen that, contrary to NN (green curves), the optical flow Recall (solid red curve) does not reach the best achievable Recall. Despite the relaxed consistency conditions (at least 2 out of 4) some good matches are not consistent with the estimated optical flows. This leads to missing some good matches. But the Precision (red solid curve) is clearly higher than the NN's one.
The presented experiment show that the optical flow driven 3D matching is an efficient way to get a high percentage of reliable matches (high Precision), even when the ppp gets close to 0.1. It clearly outperforms the Precision of the NN.

5. Subvoxel refinement

The optical flow driven 3D matching step yields reliable matchings, most of the ghosts have been discarded. But the remaining particles are located on a voxel grid (v/p=1/2). Accurate motion estimation requires subvoxel localisation of the particles, the more so the displacement field is small compared to the voxel size. In classical PTV, up to ppp=0.001, accurate 3D localisation is achieved through the establishment of accurate correspondences involving subpixel localisation of the particles in the images and epipolar geometry (Maas et al., 1993).
With the considered ppp, this is no longer possible. On one hand, accurate subpixel localisation is prevented by the mostly overlapping particles; on the other hand, with 4 cameras, the number of ambiguities is too large for a robust solution of the correspondence problem (Maas et al. 1993).
Due to the impossibility to extract accurate subpixel location for the particles, one can't formulate the subvoxel refinement as the minimization of the distances between the subpixel location in the image and projections of the 3D particles'coordinates.
Instead, using the PVR model, $Y(\boldsymbol{x}) = \sum_p E_p h\big(\boldsymbol{x} - F(\boldsymbol{X_p})\big)$ we formulate the subvoxel refinement as the global minimization of the difference between observed images and predicted ones. The predicted images depend on the 3D positions $\boldsymbol{X_p}$ and intensities E_p of the particles which are the variables to optimize. The optimization problem can be formulated as the minimization of the following criteria:

$$\min_{E_p, \boldsymbol{X_p}} \left\{ \sum_j \sum_{\boldsymbol{x}} \sum_p \left\| Y_j(\boldsymbol{x}) - E_p h(\boldsymbol{x} - F_j(\boldsymbol{X_p})) \right\|^2 \right\}$$

Where Yj are the observed images, $\boldsymbol{x}$ is any location in image plane, F_j is the projection function in image j and h is the PSF function. Without loss of generality and to alleviate the notation we suppose that h do not depend on image j.

The PSF is compactly supported; the optimization can be performed on neighbourhood of the particles projections. Let $V_j(p)$ be a neighbourhood of the projection of particle p in image j. eg. a 3x3 or 5x5 window centred on the initial particle projection. The optimization problem boils down to:

$$\min_{E_p,\, \boldsymbol{X_p}} \left\{ \sum_p \sum_j \sum_{\boldsymbol{x} \in V_j(p)} \left\| Y_j(\boldsymbol{x}) - E_p h\left(\boldsymbol{x} - F_j(\boldsymbol{X_p})\right) \right\|^2 \right\} \quad \text{(Equation 3)}$$

Given that we look for subvoxel refinement, we suppose that the neighbourhood of each particle is given by the initial projection and do not change during the optimisation process.

It is worth to mention that contrary to (Wieneke, 2013) and (Schantz et al., 2013), that iteratively optimize the particles' location one by one, we perform the optimization globally to fully account for the interactions between particles. The evaluation of the two optimisation strategies, local versus global, is an on-going work.

5.1 Optimisation procedure

If the number of particles is low, say smaller than 20000, we perform the optimization all at once. We derive the analytical expression the jacobian of the criteria given by equation 4 and use the Levenberg-Marquardt algorithm of Matlab to perform the optimisation.
When the number of particles is greater and consequently when the number of variables exceeds 4x20000, this is no more tractable, at least using our 8 core, 64G ram workstation, with the Matlab *lsqnonlin* procedure. The next section explains how such a high dimension problem can be tackled.

5.2 Graph partitioning

When the ppp grows, the optimization of the above criteria with a number of variables 4 times the number of particles may become intractable. The solution is to cut out the problem in smaller parts, but taking into account, as much as possible, particles' interactions.
This can be understood considering a toy example with 4 images, 6 particles projected in the images with 5x5 neighbourhood (see Fig. 5). Two particles interact if their neighbourhoods intersect. It can be seen Fig. 5 that there are 3 independent sets of interacting particles: {1,2,3} {4,5} and {6}. These are the connected components of the graph. A particle of one set has no interactions with the particles of the two other sets. Consequently, each connected component can be optimized independently of the others. These sets are the connected components of the graph defined by the neighbourhood intersection relationship.

Generalizing, we say that two particles p and q interact if for some image j their neighbourhoods intersect: $V_j(p) \cap V_j(q) \neq \emptyset$. This relationship defines a graph whose connected components can be optimized independently.

A graph can be represented by an adjacency matrix A, where the entry A(i,j) is the weight of the edge that links vertices “i” and “j”. For our toy example Fig. 5, it counts the number of interaction of particle “i” and “j”. The adjacency matrix of the toy example is given Fig. 5. One can see the “block diagonal” structure with the 3 connected components.

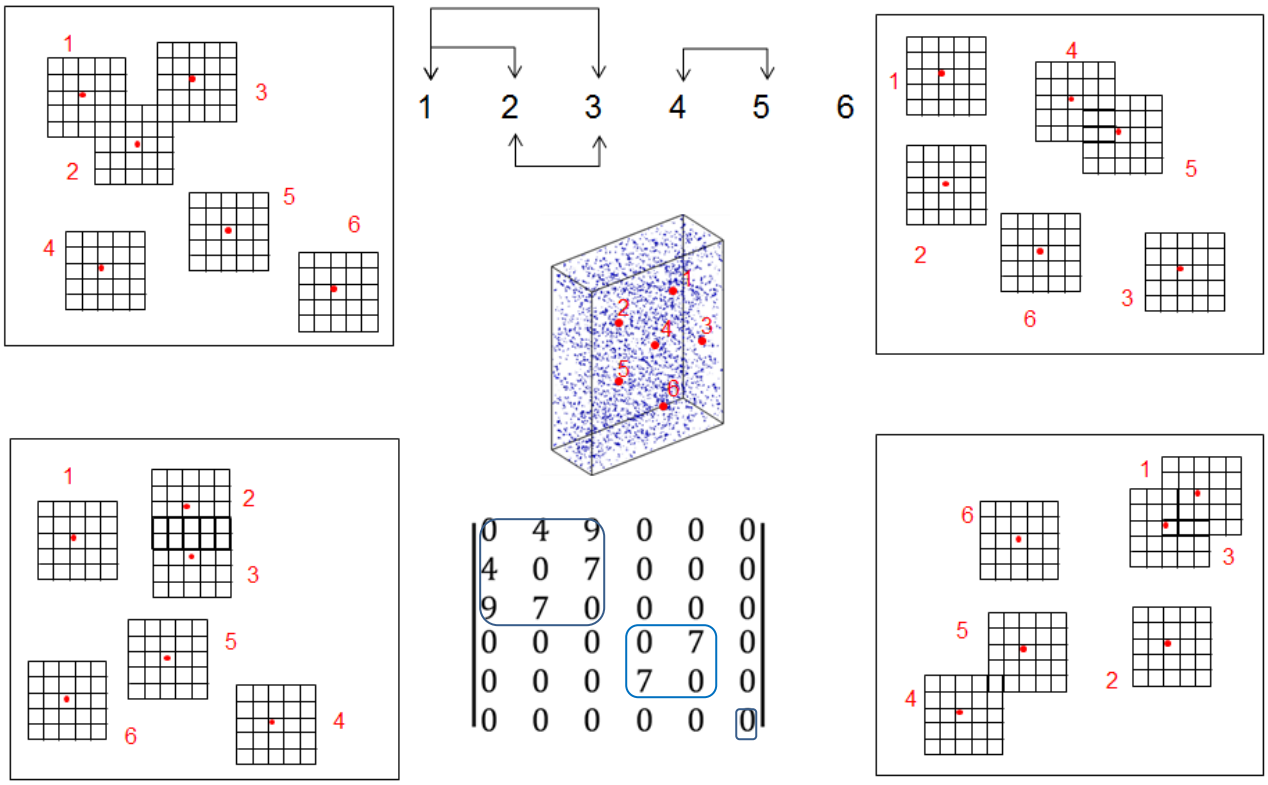

Fig. 5: Toy example with 6 particles. There are 3 connected components {1,2,3} {4,5}, {6}. The adjacency matrix describes the graph.

Unfortunately, very quickly when the ppp is greater than 0.02 the graph with four images is entirely connected, ie. the whole graph is made of only one connected component. Transforming a connected graph into several connected components cannot be done without cutting edges. Let's consider the following connected graph (Fig. 6):

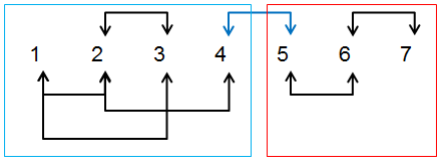

Fig. 6: Cutting the edge linking 4 and 5 yields 2 connected components.

Cutting the edge linking vertices 4 and 5 yield 2 connected components {1 2 3 4} and {5 6 7}

The aim of Graph Partitioning, which is a very active research area, is to partition a connected graph into several connected components minimizing either the number of cut edges, or the sum of the weight of the cut edges. METIS (Karypis et al.) is a set of serial programs for partitioning graphs, partitioning finite element meshes, and producing fill reducing orderings for sparse matrices. METIS provides a powerful algorithm to partition a graph given its adjacency matrix.

5.3 Optimisation procedure for high dimension sets

We compute the adjacency matrix of the graph for a fixed neighbourhood. We use METIS (Karypis et al.) to get connected components with a maximum size of about 20000 particles and we use the Levenberg-Marquardt algorithm of Matlab to perform the optimisation independently on each partition of the graph.

6. Performance measures for the vector field

Displacement fields obtained by the 3D matching of the particles are then assessed using, firstly, the averaged norm of error between estimated and ground truth displacement of the detected particles, defined as:

$$Disp_{part} = \frac{1}{Nmatch} \sum_{i=1}^{Nmatch} \left\| \hat{d}(\hat{p}_i) - d_{gt}(\hat{p}_i) \right\|$$

Where N_{match} is the number of matched particles, $\hat{p}_i$ is the estimated location of a matched particle, $\hat{d}(\hat{p}_i)$ is the displacement given by the matching and $d_{gt}(\hat{p}_i)$ is the true displacement at $\hat{p}_i$ location.

Secondly by a measure on a grid where the displacement is interpolated inside the convex hull of the particles:

$$Disp_{grid} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{d}(\boldsymbol{X}) - d_{gt}(\boldsymbol{X}) \right\|$$

Where N is the number of points on the grid, $\boldsymbol{X}$ is any location on the grid, $\hat{d}(\boldsymbol{X})$ is the estimated displacement and $d_{gt}(\boldsymbol{X})$ is the true displacement. These two measures are given in v/p=1 voxels unit.
It should be stressed that no precaution is taken to linearly interpolate the displacement on the grid, ie. no filtering of the wrong vectors.

6.1 Synthetic experiment: Vortex ring

The reconstruction (LocM-CoSaMP) is performed with a voxel to pixel ratio equals to 0.5. The interpolation grid 158x167x41 is defined in mm as follows: [-41.7 41.7] x [-39.3 39.3] x [-10.00 10.00] by step of 0.5 mm (ie. 5 voxels). This domain is included in the polyhedron seen by all the cameras. Outside of this domain there is virtually no movement. That is the reason why we restrict the performances to this domain. They would have been better if we had computed them on the whole commonly seen polyhedron.

Fig. 7 shows the performance measures $Disp_{part}$ and $Disp_{grid}$ as a function of the ppp on a voxel and subvoxel basis. It can clearly be seen the improvement achieved by the subvoxel refinement. The subvoxel mean error $Disp_{part}$ is up to 5 times lower than the voxel one for the lowest ppp with a value of 0.07 voxels. When considering the interpolation on the grid, the subvoxel mean error $Disp_{grid}$ is 2.6 times lower for the lowest ppp than its voxel counterpart.

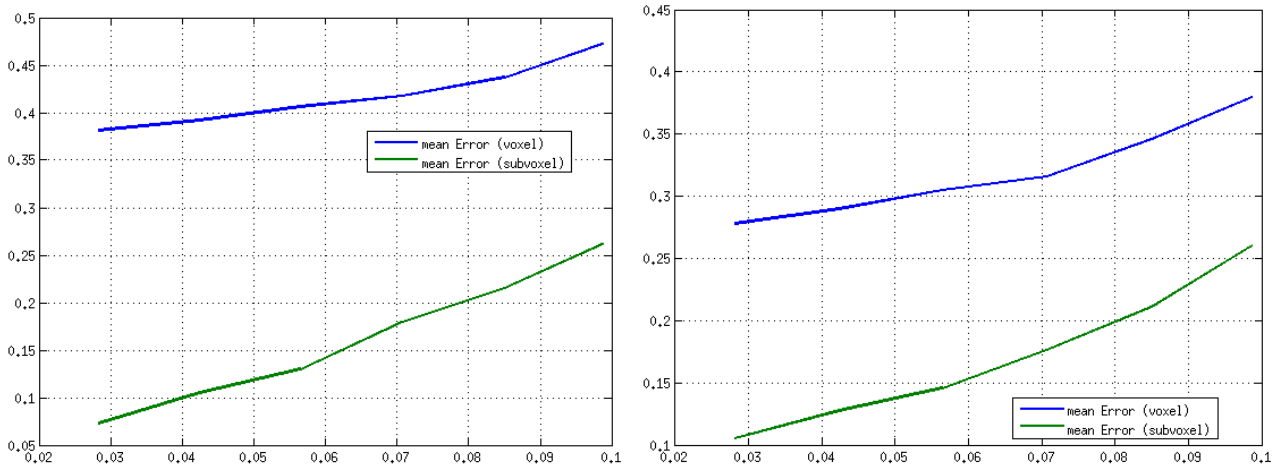

Fig. 7: Left hand side, $Disp_{part}$ mean error as a function of the ppp in v/p=1 voxel unit. Without refinement (blue), with refinement (green). Right hand side, $Disp_{grid}$ mean error on the grid.

6.1.1 Interpolation on a grid

It should be stressed again that the shown interpolations are performed "as is", without averaging on an interrogation window as done in (Schantz et al. 2013) and without any attempt to reduce the influence of wrong vectors. The interpolation is linear and wrong vectors may perturb locally the result.
The displacement is given in v/p=1 voxel units. (U,V,W) refer to the three components of the displacement on axis (X,Y,Z) of the world frame. We present results for ppp=0.028 and ppp=0.047. The illustrations shown are slices for various planes of the interpolated vector field (U,V,W) on the grid. The general layout of Fig. 8 to Fig. 10 and Fig. 11 to Fig. 13 is as follows. On the left hand side, the ground truth, in the middle the voxel based interpolation and on the right hand side the subvoxel refinement based interpolation. The same comment applies to all these figures. The structure of the flow is well captured and improvements achieved by the subvoxel refinement are clearly visible. It can also be seen that the subvoxel refinement can help to get rid of a few remaining wrong matches.
Indeed, it may happen that during the optimisation some particles move far away (i.e. several tens of voxels) from their initial positions. If so, the two corresponding matched particles are removed. We have observed that ghosts were involved in this process. This topic is currently under inspection.
Hereafter we do not deal with the relevant question of determining which ppp should be used to better describe a complex flow with our method. This study remains to be done. We only provide these illustrations to give an overview of how well the interpolation of the sparse vector field for commonly used ppp in real experiments can capture the structure of the flow.

6.1.1.1 ppp=0.028

The $Disp_{grid}$ (rmse) measure equals 0.105 voxels. Fig. 8 shows W in plane Z=0. Although for this ppp they are few wrong vectors in the volume, the influence of one of them can be seen on the outer right limit of the vortex.

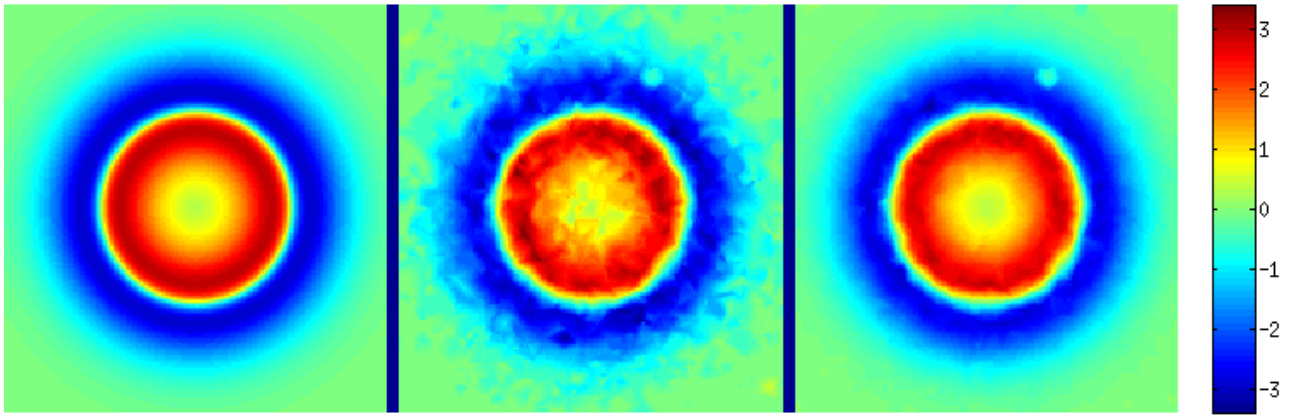

Fig. 8: ppp=0.028. W in plane Z=0. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

Fig. 9 shows U in plane Z=-5.5 mm. It can be seen that a wrong vector disturbs the voxel based interpolation. It is corrected or removed by the subvoxel refinement, as explained above. Fig. 10 shows V in plane Z=-5.5 mm.

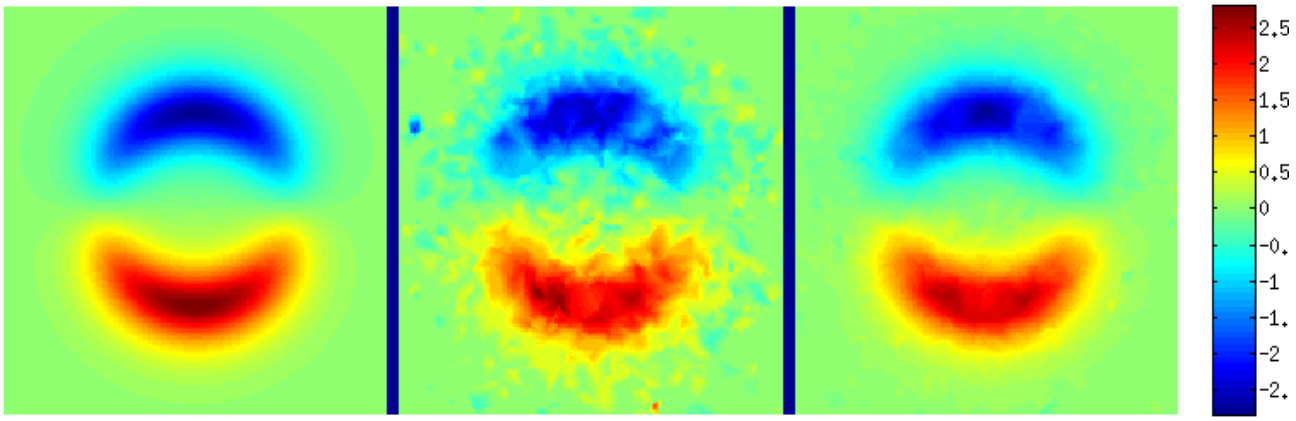

Fig. 9: ppp=0.028. U in plane Z=-5mm. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

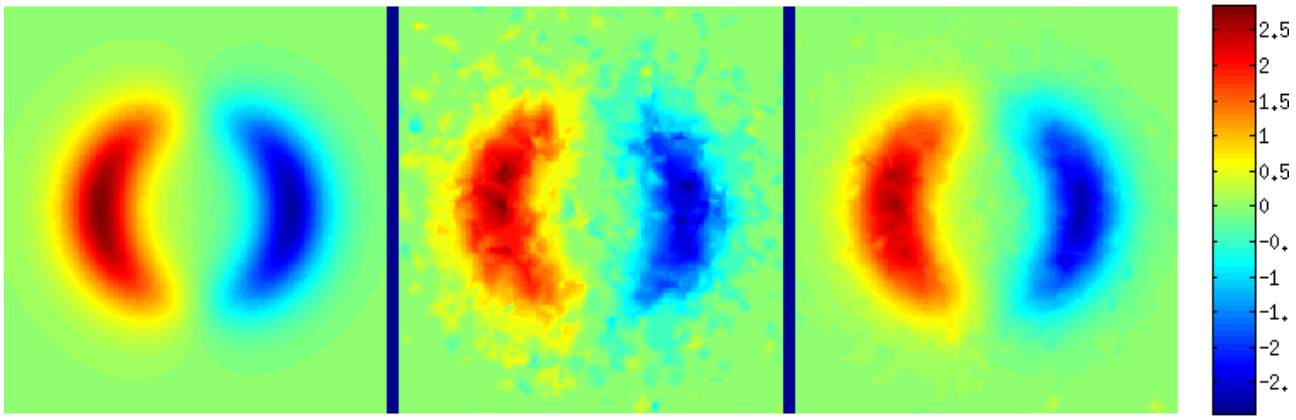

Fig. 10: ppp=0.028. V in plane Z=-7.5mm. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

6.1.1.2 ppp=0.047

For this ppp the $Disp_{grid}$ measure equals 0.127 voxels. Fig. 11 illustrates W in plane Z=0. The subvoxel interpolation is quite satisfactory.

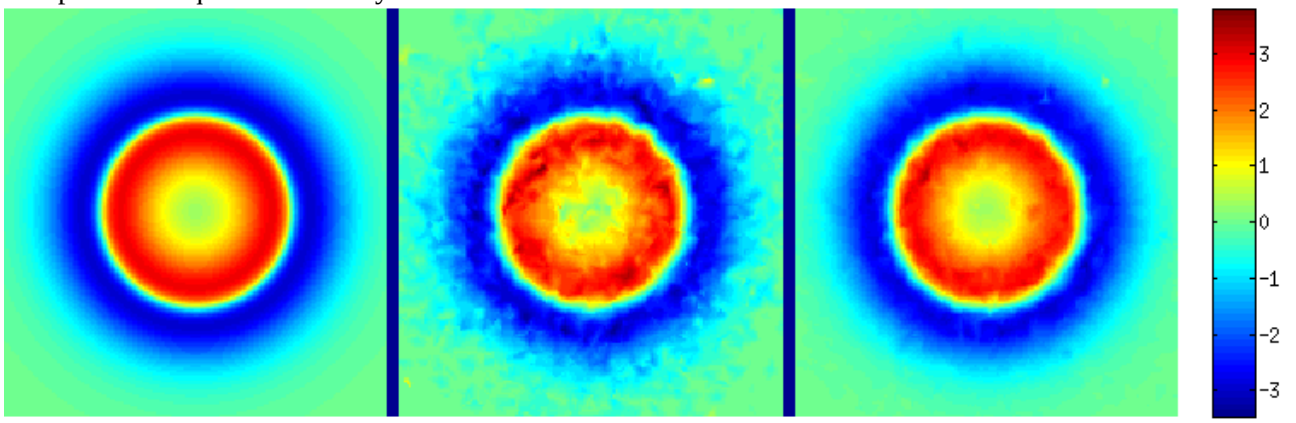

Fig. 11: W in plane Z=0. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

Fig. 12 shows U in plan Z=-5.5 mm. It can be seen that although the refinement eliminates some wrong vectors, at least two remain and locally disturb the interpolation.

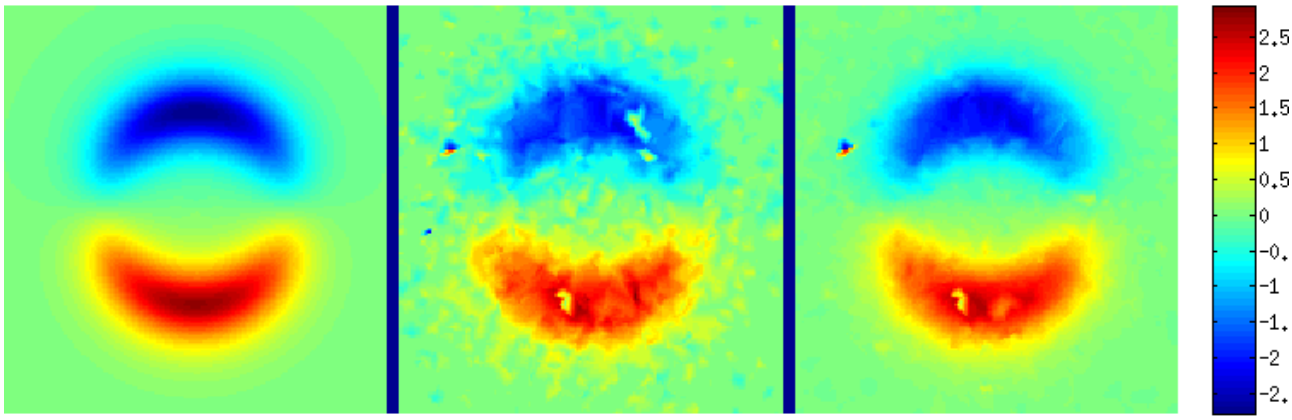

Fig. 12: U in plane Z=-5.5mm. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

Fig. 13 shows V in plane Z=-8 mm. Several wrong vectors are eliminated by the subvoxel refinement.

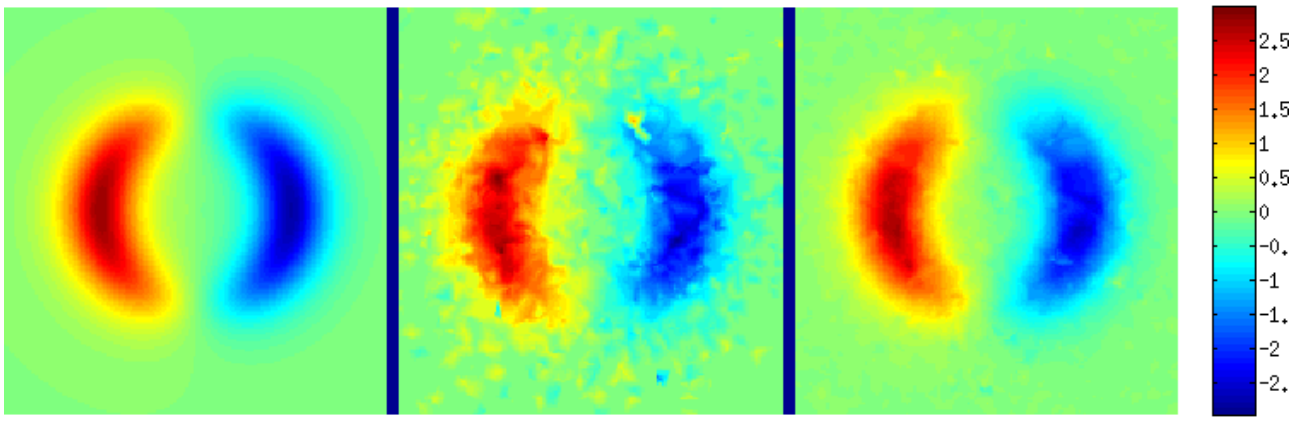

Fig. 13: V in plane Z=-8 mm. (Left) Ground truth – (Middle) voxel based – (Right) subvoxel refinement

7. Discussion

We have presented what may be described as a credible alternative to conventional 2 frames 4 cameras Tomo-PIV - at least until ppp=0.05- with many benefits. The particles are only represented by four real numbers - 3 for position coordinates, one for intensity - saving a huge amount of memory. Starting from an ultra sparse tomographic reconstruction "one voxel-one particle" with coarse location, our method provides accurate subvoxel 3D location of the particles at the two time steps along with reliable matches, ie. it follows the motion of the particles between time t and t+dt. Consequently, vector field computation does not rely on 3D cross correlation. It avoids thus the drawbacks of large interrogation windows. One of the most salient features of our approach is that it does not rely on previously estimated kinetic parameters and start from scratch. The 3D matching is driven by the optical flow measured in the images. The matches are reliable and most of the ghosts are filtered. We have shown on a vortex ring synthetic experiment that the instantaneous flow field may be interpolated on a regular grid without taking any special precaution to filter wrong vectors, at least until ppp=0.05. Illustrations show how well the structures of the flow are captured.

It should be stressed again that in the presented illustration there is no mechanism to filter wrong vectors. Such a scheme should be very beneficial and should alleviate their influence on the interpolation and rmse.

Looking at Fig. 12 one might think that lower ppp than those used in our experiments would give better rmse results. It is true for the $Disp_{part}$ measure, but below a certain ppp $Disp_{grid}$ is expected to rise because interpolation requires a minimum number of measures.

Thorough evaluation against tomo-PIV is an ongoing work and will be dealt in a future paper. One can already remark that the rmse computed on the grid in a ppp range of [0.03 0.06] compares favorably with Tomo-PIV results on the same case (Champagnat et al. 2014).

Our scheme is designed to be conservative. It is deemed better not to match two particles rather than risk an

erroneous match. It is both its strength and its weakness. A high Precision is favored, but it is at the expense of the Recall. Currently, long term tracking cannot be associated with high Recall. Fig. 4 shows that for the vortex ring experiment the Recall for two time steps is around 0.9 for ppp≤0.05. A first order approximation would say that for n time step it would be 0.9^{n-1}. The drop is rather quick when n grows. Long tracks are not very likely.

Although some good matches may not meet the optical flow requirement because this latter may not be perfectly estimated, the main limiting factor is that the reconstruction step does not detect all the particles in the volume. Increasing the sparsity parameter yields more true particles in the reconstruction and more good matches (Recall grows) but it results also in more ghost and wrong matchings (Precision drops). The left and right side of Fig. 14 give two insights of this. The normalized sparsity (NS) is the ratio of the sparsity parameter and the number of particles. The left hand side of Fig. 14 shows for the vortex ring experiment (ppp=0.057), Precision and Recall as a function of the normalized sparsity. The right hand side shows Precision and Recall as a function of the fraction of true particles detected in the first time step reconstruction. It can be seen on both side that up to some breakpoint (NS=1; fraction of detected particles ≈ 0.95) the Precision is almost steady while the Recall soars. Above this breakpoint, the Recall does not grow much while the Precision falls very quickly. It justifies our choice of using NS equals 1.

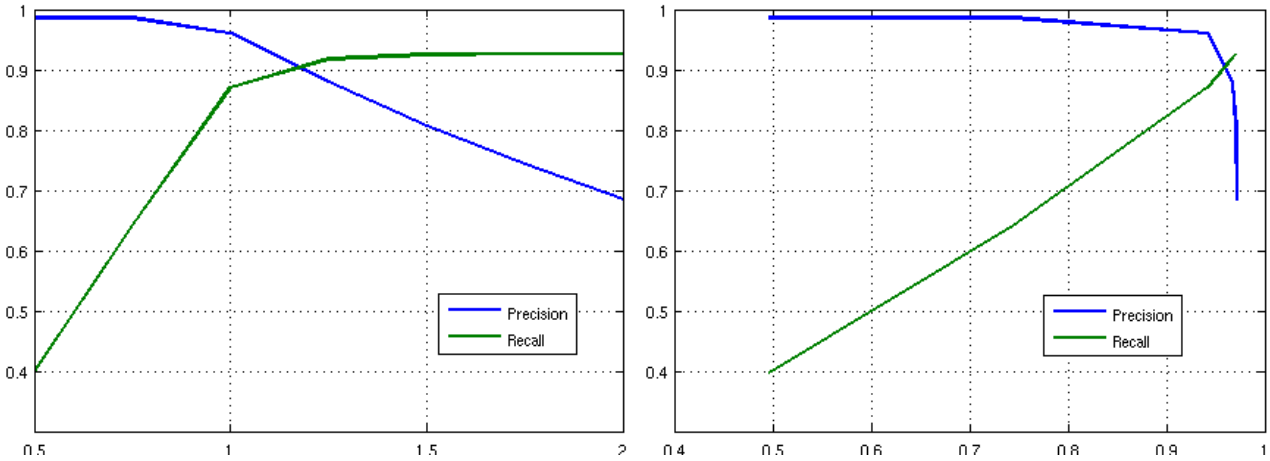

Fig. 14: Vortex ring ppp=0.056. Left, Precision and Recall as a function of the normalized sparsity. Right: Precision and Recall as a function of the fraction of true particle detected in the first time step.

This leads to the conclusion that our approach alone cannot be used to produce a high fraction of long tracks. To reach this goal, there are several possible solutions that may be implemented synergistically:

- Strap down with a prediction mechanism based on kinetic to increase the Recall without affecting the Precision.
- Tracklet (ie. short tracks) stitching through the use of instantaneous interpolated vector field is also a promising approach.
- Use the almost sure matched particles to revert to the reconstruction and refine it by subtracting these particles from the images.

However, it should be stressed that given that the acceleration of a particle can be computed with 3 time steps, according to the 0.9^{n-1} rule our method can yield acceleration for at least 80% of the particles in the volume and yield a reliable acceleration map.

Acknowledgements

The authors would like to thank Dr.Riadh Fezzani for interesting discussions on graph partitioning and for providing them with tools and references on this topic.

Reference

Atkinson C., Soria J (2009). An efficient simultaneous reconstruction technique for tomographic

particle image velocimetry. Experiments in Fluids 47, 553–568.
Champagnat F., Cornic P., Cheminet A., Leclaire B., Le Besnerais G. (2013). Tomographic PIV: particles vs blobs, in 'Proceedings of PIV13'.
Champagnat F., Cornic P., Cheminet A., Leclaire B., Le Besnerais G., Plyer A. (2014) Tomographic PIV: particles vs blobs. Submitted to MST.
Champagnat F., Plyer A., Le Besnerais G., Leclaire B., Davoust S., Le Sant Y. (2011). Fast and accurate PIV computation using highly parallel iterative correlation maximization. Experiments in Fluids 50, 1169–1182.
Cheminet A., Leclaire B., Champagnat F., Cornic P., Le Besnerais G. (2013). On factors affecting the quality of tomographic reconstruction, in 'Proceedings of PIV13'.
Cornic P., Champagnat F., Cheminet A., Leclaire B., Le Besnerais G. (2013). Computationally efficient sparse algorithms for tomographic PIV reconstruction, in 'Proceedings of PIV13'
Elsinga G. E., Scarano F., Wieneke B. van Oudheusden B. (2006). Tomographic particle image velocimetry Experiments in Fluids 41, 933–947.
Grégoire G (2011). Introduction aux processus ponctuels spatiaux. http://www.youscribe.com
Karypis G. et al. METIS. http://glaros.dtc.umn.edu/gkhome/lab
Maas H. G., Gruen A., Papantoniou D. (1993). Particle tracking velocimetry in three-dimensional flow (Part I). Experiments in Fluids 15, 133-146.
Malik N. A., Dracos Th., Papantoniou D. A. (1993). Particle tracking velocimetry in three-dimensional flow (Part II). Experiments in Fluids 15, 279-294.
Ouellette N. T., Xu Haitao (2006). A quantitative study of three-dimensional Lagrangian particle tracking algorithms. Experiments in Fluids 40, 301-313.
Needell D., Tropp J.A. (2009) CoSaMP: Iterative signal recovery from incomplete and inaccurate sample. Appl. Comput Harmon. AnalVol 26 n°3 pp 301-321.
Novara M., Batenburg K.J.,Scarano F. (2010). Motion tracking-enhanced MART for tomographic PIV. Meas. Sci. Technol. 21
Schantz D. , Schröder A., Gesemann S., Michaelis D.,Wieneke B. (2013). Shake the box: A highly efficient and accurate Tomographic Particle Tracking Velocimetry (TOMO-PTV) method using prediction of particle positions, in 'Proceedings of PIV13'.
Schroder et al. (2009) Lagrangian aspects of coherent structures in a turbulent boundary layer flow using TR-Tomo PIV and PTV blobs, in 'Proceedings of PIV09'.
Shroder et al. (2011) Eulerian and Lagrangian views of a turbulent boundary layer flow using time-resolved tomographic PIV. Exp. Fluids 50.
Wieneke B. (2011). Iterative reconstruction of volumetric particle distribution, in 'Proceedings of PIV11'.
Wieneke B. (2013). Iterative reconstruction of volumetric particle distribution. Meas. Sci. Technol. 24.