SYNCHRONOUS ORTHOGONAL DYNAMIC PROGRAMMING FOR PARTICLE IMAGE VELOCIMETRY

Georges Quénot[1]

Keywords: *Stereo PIV, Optical Flow, Dynamic Programming*

ABSTRACT

This paper presents an improvement of the Orthogonal Dynamic Programming optical flow technique that permits a better use of all the available information during the iterative search of the vector field in the case of stereo PIV. The two-component apparent vector fields are computed simultaneously (synchronously) for all available viewpoints. Coherency constraints issued from the geometry of the experimental setup and based upon the existence of a global three-component vector field are applied at each global step of the iterative search. Tests carried out using the Standard Images from the Visualization Society of Japan shows a small improvement in accuracy and a significant improvement in robustness.

1 INTRODUCTION

Many iterative methods have been proposed for the implementation of Particle Image Velocimetry (PIV) systems [1] [2] [3] [4]. They usually yield a significant increase in accuracy and robustness through the use of variable size search windows (hierarchical / multi-resolution approaches) and image grid deformation (subpixel alignment search). These methods (like the classical ones) permit the recovery of the two in-plane velocity components in a planar section of a 2D or 3D fluid flow illuminated by a thin laser sheet from a single image sequence taken from a single viewpoint.

Using image sequences taken simultaneously by two or more cameras from different viewpoints, it is possible to recover the three velocity components, again in a planar section of a 3D flow field illuminated by a thin laser sheet. The classical way to do this is to compute the apparent two-component vector field independently for the two (or more) image sequences and then to combine these two (or more) two-component vector fields (defined in the camera focal planes in image coordinates) into a single three-component vector field (defined in the laser sheet plane in world coordinates) using the camera calibration parameters (generally using a least square method).

We may notice that the two (or more) two-component vector fields that are iteratively searched for independently for each image sequence are actually *not* independent from each other. Instead,

Author(s): [1]*CLIPS-IMAG, BP 53, 38041 Grenoble Cedex 9, France*

Corresponding author: Georges Quénot

they are constrained among them by the geometry of the experimental setup and the existence of a single three-component vector field of which they are the projection. The purpose of this work is to investigate the possibility to get an improvement in the accuracy and/or robustness of iterative methods for the three-component vector field search by exploiting the fact that the two (or more) two-component vector fields that are classically searched for independently are actually constrained among them in such a way.

The idea is implemented and evaluated here using the Orthogonal Dynamic Programming Optical Flow technique for the iterative search of two-component vector fields. However, it should be applicable to any other iterative technique that offers suitable characteristics (mainly, the full density and continuity of vector fields).

2 SYNCHRONOUS ITERATIVE VECTOR FIELD SEARCH

2.1 Principle

The final building of the three-component vector field makes use of the camera calibration data as additional constraints. During this process, data extracted from the different viewpoints are merged in a way that not only reduces their volume but, additionally, make them coherent with respect to the geometrical constraints of the experimental setup. This coherency can be enforced when the three-component vector field (defined in the world coordinates) are back-projected (split) into two (or more) two-component vector fields (defined in camera image coordinates). When this is done, the resulting two-component vector fields become coherent among them with respect to the geometrical constraints of the experimental setup. This is unlikely to be the case for the directly and independently extracted two-component vector fields: though it should *roughly* be the case from the experimental system design and calibration, this is not exactly the case since errors are introduced during the vector field extraction and these errors are not (or, at least, not fully) correlated. During the merge and back-projection of the vector fields, errors are made coherent with the rest of the vector fields and also averaged, which implies statistically reduced.

The basic idea in this work is that, if the vector field extraction process is based on an iterative refinement, instead of being done only once at the end of the process (Figure 1), the merge step (completed by a back-projection split step) could be inserted at each iteration (Figure 2). This ensures that the refined vector field is kept coherent with the geometrical constraints of the experimental setup during the whole process and not only made so at the end. It is expected that the residual error, which is averaged during the merge and back-projection step, will be statistically reduced during the whole process leading to more accurate and more robust results.

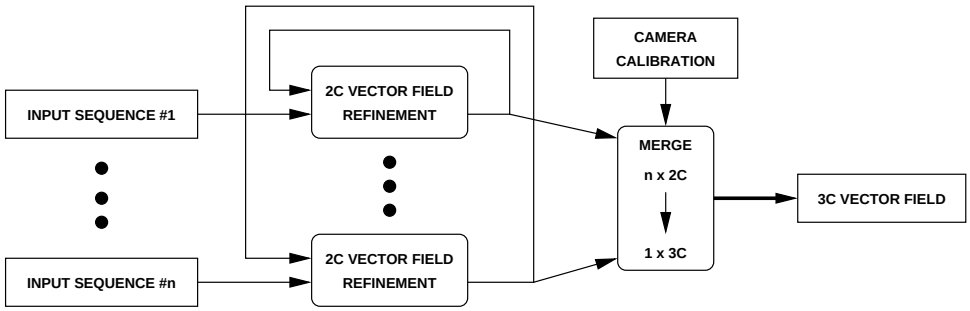


Fig. 1 Classical iterative vector field search

Synchronous Iterative Vector Field Search emphasizes the fact that the vector field refinement iterations that have to be carried out for the recovery of the two-components vector fields for

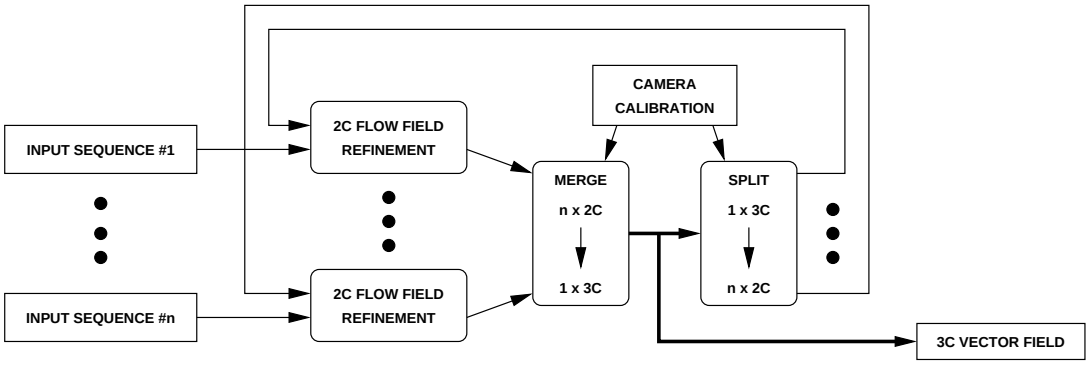


Fig. 2 Synchronous iterative vector field search

each of the viewpoints are no longer executed independently but synchronously in the sense that corresponding iterations are executed as a group and that between each group execution, a synchronization step is inserted. The synchronization step here is "simply" the merge of the two (or more) two-components vector fields into a single three-component vector field (using the geometrical constraints of the stereo PIV experimental setup and a least square minimization algorithm) and the back-projection of this three-component vector field into two (or more) two-components vector fields.

2.2 Merge of the two-component vector fields

For each viewpoint c, the useful geometrical constraints can be expressed by the f_c and g_c functions:

$$(x,y) = f_c(u_c,v_c) \qquad (k_x,k_y) = g_c(u_c,v_c) \tag{1}$$

where (u_c,v_c) are the coordinates of a point in the image focal plane of camera c, $(x,y,0)$ are the coordinates of the corresponding location in the illuminated plane, and $(k_x,k_y,1)$ are the components of the director vector of the line passing through the $(x,y,0)$ point and projecting (in the neighborhood of this point) on the (u_c,v_c) point in the image focal plane of camera c. These functions can be expressed explicitly using the camera calibration parameters (and, optionally, informations about changes in the refraction index along the optical pathway) using ray tracing approaches. Figure 3 shows an example using only the X and Z components.

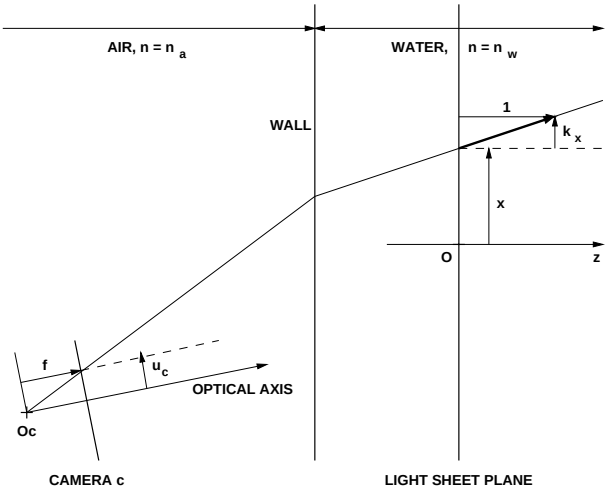


Fig. 3 Ray tracing from a camera focal plane location

The merge of the two-component vector fields is done in two steps. In the first one, vector fields obtained in the camera focal planes (in camera coordinates (u_c, v_c) and units (pixels/timestep) are forward-projected on the light sheet plane (in world coordinates $(x, y, 0)$ and units (i.e. cm/s). For this, the f_c^{-1} functions, inverse of the f_c functions have to be used. Except in the case in which there is no change of the refraction index along the optical pathway, the f_c^{-1} functions cannot be expressed explicitly but they can be computed practically as the inverse of the f_c functions. The $(v_{xc}(x,y), v_{yc}(x,y))$ vector field, forward-projection of the camera c vector field $(v_{uc}(u,v), v_{vc}(u,v)$ can then be computed as:

$$\begin{pmatrix} v_{xc}(x,y) \\ v_{yc}(x,y) \end{pmatrix} = \frac{1}{t_s}.(\mathbf{grad}\ f_c)(f_c^{-1}(x,y)).\begin{pmatrix} v_{uc}(f_c^{-1}(x,y)) \\ v_{vc}(f_c^{-1}(x,y)) \end{pmatrix} \quad (2)$$

t_s being the time step. For this computation be practically possible, it is necessary that a (v_{uc}, v_{uc}) can be computed at any (even non-integer) (u,v) location in the camera c focal plane. This requires that the recovered vector fields be dense enough and continuous enough so that interpolation techniques can be usable and accurate. The forward projections are computed independently for each viewpoint c.

The second step of the merge of the two-component vector fields is the building, in the light sheet plane, of the single three-component vector field. At a given location (x,y), a (v_x, v_y, v_z) three-component velocity vector is projected for each viewpoint c along the $(1, k_{xc}, k_{yc})$ associated director vector ($(k_{xc}, k_{yc}) = g(f_c^{-1}(x,y))$) into a (v_{xcp}, v_{ycp}) vector as:

$$\begin{pmatrix} v_{xcp} \\ v_{ycp} \end{pmatrix} = \begin{pmatrix} 1 & 0 & -k_{xc} \\ 0 & 1 & -k_{yc} \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} \quad (3)$$

The goal is to find a (v_x, v_y, v_z) vector whose projections (v_{xcp}, v_{ycp}) best matches the (v_{xc}, v_{yc}) vector fields, forward-projections of the (v_{uc}, v_{vc}) extracted vector fields. A natural choice is the (v_x, v_y, v_z) vector that minimizes the residual quadratic error:

$$E(v_x, v_y, v_z) = \sum_c \left((v_{xcp} - v_{xc})^2 + (v_{ycp} - v_{yc})^2 \right) \quad (4)$$

It immediately comes from equations 3 and 4 that $E(v_x, v_y, v_z)$ is a quadratic function of v_x, v_y and v_z. Its minimum value is obtained when its three partial derivatives $\partial E/\partial v_x$, $\partial E/\partial v_y$ and $\partial E/\partial v_z$ are simultaneously null. These partial derivatives are linear functions of v_x, v_y and v_z and, if there is at least two different viewpoints, they form a rank 3 linear system that has a unique (v_x, v_y, v_z) solution. The $(v_x(x,y), v_y(x,y), v_z(x,y))$ vectors are computed independently for each desired (x,y) location, typically a 2D-grid in the light sheet plane.

2.3 Back-projection of the three-component vector field

The back-projection is much simpler and computed independently for each (u_c, v_c) location in the focal plane of each camera c. The corresponding (x,y) location and (k_x, k_y) director vector components are computed using the f_c and g_c functions (equation 1). The corresponding $(v_x(f_c(u_c,v_c)), v_y(f_c(u_c,v_c)), v_z(f_c(u_c,v_c)))$ three-component velocity vector is then interpolated from the nearest available $(v_x(x,y), v_y(x,y), v_z(x,y))$ values (since, the $f_c(u_c,v_c)$ location is usually out of the grid locations for which the three-component vectors were computed). This three-component vector is then projected on the laser sheet plane along the director vector associated to each of the viewpoints, it is simply done using equation 3 but here with $(k_{xc}, k_{yc}) = g_c(u_c, v_c)$. Finally, the $(v_{xcp}(f_c(u_c,v_c)), v_{ycp}(f_c(u_c,v_c)))$ obtained vector is back-projected on the focal plane using the following equation, symmetrical of equation 2:

$$\begin{pmatrix} v_{uc}(u_c, v_c) \\ v_{vc}(u_c, v_c) \end{pmatrix} = t_s.(\mathbf{grad}\ f_c)^{-1}(u_c, v_c).\begin{pmatrix} v_{xcp}(f_c(u_c, v_c)) \\ v_{ycp}(f_c(u_c, v_c)) \end{pmatrix} \quad (5)$$

2.4 Interpolation of vector fields

Generally, each two-component vector field is defined on a discrete rectangular grid aligned to the corresponding pixel image grid and possibly subsampled from it. Generally also, the three-component vector field is defined in the laser sheet plane also on a discrete rectangular grid aligned with the chosen cartesian real world coordinates (possibly up to an isometry). There is little chance that there is a good correspondence between these different grids through the forward and backward projections along the optical pathways. Therefore, in both the merge and split steps, it is necessary to transform vector fields defined into "their" grids into vector fields defined into other grids. In order to keep the resolution of the vector fields during the merge and split synchronization step, it is necessary that the density (total number of points) of all the grid be roughly the same. Vector values for non-integer grid coordinates can be obtained by bilinear interpolation (or any other method). This requires that the used iterative vector field search technique provides dense (without holes) and "reasonably" continuous vector fields.

The double interpolation required for the merge and split induces some smoothing of the vector fields. This is not necessarily a problem for iterative methods and may even improve the result by enforcing some local coherency.

It is theoretically possible to evaluate the vector fields in the image sequences at locations that are the back-projections in the camera focal planes of the grid used for the definition of the three-component vectors. This would eliminate the need for vector field interpolation both forward and backward. However, this would require the iterative computation of the two-components vectors at non-integer and non-aligned locations and this is not how most iterative two-components vector field evaluation methods currently work. This possibility might be used in the future.

3 SYNCHRONOUS ORTHOGONAL DYNAMIC PROGRAMMING

The Orthogonal Dynamic Programming (ODP) optical flow computation technique [5] is a good candidate to test for the Synchronous Iterative Vector Field Search improvement since it is based on an iterative search for a dense velocity field that minimizes the L_1 or L_2 (Minkowski) distance between two (or more) images while enforcing continuity and regularity constraints. Its originality is that it transforms the search problem for a global two-dimensional alignment into a selected sequence of search problems for semi-global mono-dimensional alignments. It has proven to be a valuable tool for Particle Image Velocimetry [2] [3] and, compared with other optical flow approaches or to the classical correlation based DPIV techniques, it offers the following advantages simultaneously:

- It can be applied simultaneously to sequences of more than two images.
- It performs a global image match by enforcing continuity and regularity constraints on the flow field (this helps in ambiguous or low particle density regions).
- It provides dense velocity fields (neither holes nor border offsets).
- Local correlation is iteratively searched for in regions whose shape is modified by the flow, instead of being searched in fixed windows (this greatly improves the accuracy in regions with strong velocity gradients).
- It is able to search for subpixel displacements.
- It is able to operate on multi-band (e.g. colour) images.

The most useful of these characteristics for the "synchronous" improvement are the continuity and full density of the recovered vector field during the whole iterative process. Both are necessary for the merge and split synchronization step to operate nicely. The combination of the Synchronous Iterative Vector Field Search with the ODP technique will be referred to as Synchronous Orthogonal Dynamic Programming (SODP).

4 PERFORMANCE EVALUATION USING THE VSJ STANDARD IMAGES

Both the original (ODP) and synchronous (SODP) methods have been evaluated on calibrated synthetic test sequences [6] made publicly available (`http://www.vsj.or.jp/piv/image3d/image-e.html`) by the Visualization Society of Japan (VSJ) using a standard protocol [7] also publicly available (`http://www-clips.imag.fr/mrim/georges.quenot/-vsj-eval/evaluation.html`). There are two test sequences, STD331 and STD337, 145- and 201-images long respectively and taken each from three viewpoints simultaneously. The performance index is the global RMS error on all three-component recovered vectors and for all possible time steps (from 144 to 139 and from 200 to 195 respectively using from 2 to 7 consecutive images for the vector field recovery).

Tables 1 and 2 display the absolute (in cm/s) and relative (as a percentage of the global RMS value of the correct field) global RMS error for the ODP and SODP methods on the STD331 and STD337 test sequences respectively, using 2, 3, 5 and 7 consecutive images. A few severely erroneous vector fields were obtained for STD331 (7/144 with 2 images, 8/143 with 3 images, 4/141 with 5 images and 1/139 with 7 images). This is due to the particle density in this series which is quite below the optimum value for the ODP method and illustrates one weakness of it which is somehow a drawback of its strength: since a global match between images is searched with continuity constraints enforced by the use of dynamic programming, either ambiguity are correctly resolved using neighborhood information (general case) or local errors are propagated globally and lead to a largely wrong vector field (when approaching the limits of operation).

STD331	3×2 images	3×3 images	3×5 images	3×7 images
ODP @	0.624 (6.64 %)	0.532 (5.66 %)	0.366 (3.89 %)	0.363 (3.86 %)
density	137/144 (95 %)	135/143 (94 %)	137/141 (97 %)	138/139 (99 %)
SODP @	0.607 (6.47 %)	0.500 (5.32 %)	0.364 (3.87 %)	0.359 (3.82 %)
density	100 %	142/143 (99 %)	140/141 (99 %)	100 %
ODP	0.959 (10.2 %)	7.843 (83.5 %)	4.200 (44.7 %)	3.760 (40.0 %)
SODP	0.607 (6.47 %)	1.680 (17.8 %)	0.645 (6.86 %)	0.359 (3.82 %)

Table 1 RMS error of ODP and SODP methods on the STD331 VSJ test sequence, with manual filtering of erroneous vector fields (top rows) and with no filtering (bottom rows).

STD337	3×2 images	3×3 images	3×5 images	3×7 images
ODP	1.203 (11.6 %)	1.014 (9.84 %)	0.823 (7.99 %)	0.776 (7.53 %)
SODP	1.047 (10.1 %)	0.992 (9.62 %)	0.835 (8.10 %)	0.808 (7.84 %)

Table 2 RMS error of ODP and SODP methods on the STD337 VSJ test sequence.

The SODP enhancement was conceived for improving the robustness of the ODP method to such global mismatches and it actually worked since the number of erroneous vector field was greatly reduced (only 1/143 with 3 images and 1/141 with 5 images remained and, for those remaining, the global error was also significantly reduced). The globally spurious vector fields are quite easy to detect visually (Figure 4) and they can be manually filtered (spurious vector

fields are removed as a whole even though a significant part of them may still be correct). Table 1 shows the results with and without manual filtering. No such global errors were found with the STD337 test sequence. When no global correction occurs, the SODP method produces little (and sometimes none) improvement of the global RMS error. Therefore, the gain is more in robustness than in accuracy. The gain in accuracy is more important (or even actually a gain) when a small number (2 or 3, and this is by large the most common practical case) of consecutive images are used to evaluate the two-component vector fields. This indicates that the synchronous method can correct errors that could alternatively be corrected using more images, even if additional images are not available.

Figure 4 shows an example where the synchronous method yields a significant improvement. Third component (Z) is visualized in place of the vertical one (Y) in the bottom row (vectors must be rotated 90 degrees backward around an horizontal axis). The global RMS error for the three-component vector fields obtained by the ODP and SODP methods are of 13.28 and 0.53 cm/s (145 and 5.79 %) respectively for a global RMS value of the correct vector field of 9.16 cm/s corresponding to an average (RMS) image displacement of 8.38 pixel/interval.

5 CONCLUSION

The Synchronous Iterative Vector Field Search has been implemented using the Orthogonal Dynamic Programming (ODP) optical flow technique as the initial iterative vector field search technique. Both the original (ODP) and synchronous (SODP) versions have been successfully applied to Stereo Particle Image Velocimetry. The synchronous version yields a small improvement in accuracy and a significant improvement in robustness. The gain in accuracy is more significant in the most common case in which a small number of consecutive images are used. Both methods have currently been tested only on synthetic calibrated image sequences but in the case of classical (non stereo) PIV, it was observed that ODP was also very good on noisy synthetic sequences and on real ones [2] [3] [7].

The Synchronous Iterative Vector Field Search has been implemented and evaluated only with the ODP optical flow technique as the initial iterative vector field search technique but it could be used with any other iterative vector field search technique provided that it produces dense (without holes) and “reasonably” continuous vecotr fields.

REFERENCES

[1] Kumar S. and Banerjee S., Development and Application of a Hierarchical System for Digital Particle Image Velocimetry to Free-Surface Turbulence, *Experiments in fluids*, 10(1998), 160-177.

[2] Quénot G.M., Pakleza J. and Kowalewski T.A., Particle Image Velocimetry using Optical Flow for Image Analysis, *8th International Symposium on Flow Visualization*, Sorrento, Italy, 1-4 sept. 1998.

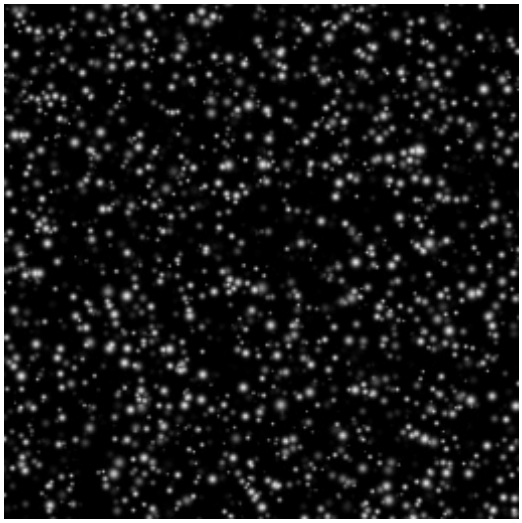
[3] Quénot G.M., Pakleza J. and Kowalewski T.A., Particle Image Velocimetry with Optical Flow, *Experiments in fluids*, 25-3(1998), 177-189.

[4] Scarano F. and Riethmuller M. L. Advances in Iterative Multigrid PIV Image Processing, *3rd International Workshop on PIV*, Santa Barbara, CA, USA, 16-18 sept. 1999.

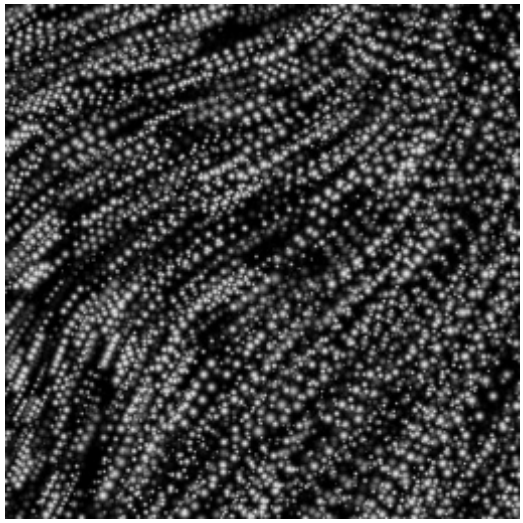
[5] Quénot G.M., Computation of Optical Flow Using Dynamic Programming, *IAPR Workshop on Machine Vision Applications*, Tokyo, Japan, 12-14 nov. 1996.

[6] Okamoto K., Nishio S., Saga T. and Kobayashi T., Standard Images for Particle Image Velocimetry, *Meas. Sci. Technol.* Vol.11, No. 6 (June 2000), 685-691

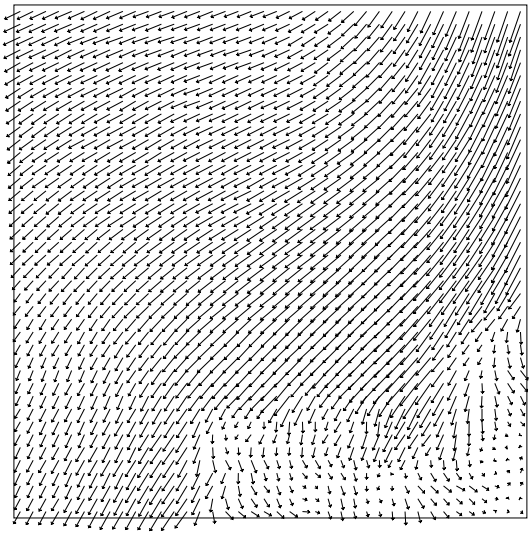
[7] Quénot G.M. and Okamoto K., A Standard Protocol for Quantitative Performance Evaluation of PIV Systems, *9th International Symposium on Flow Visualization*, Edinburgh, Scotland, UK, 22-25 aug. 2000.



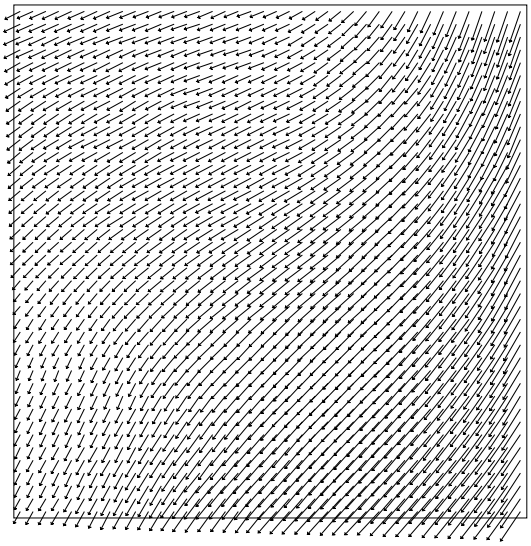
Particle image (center view)



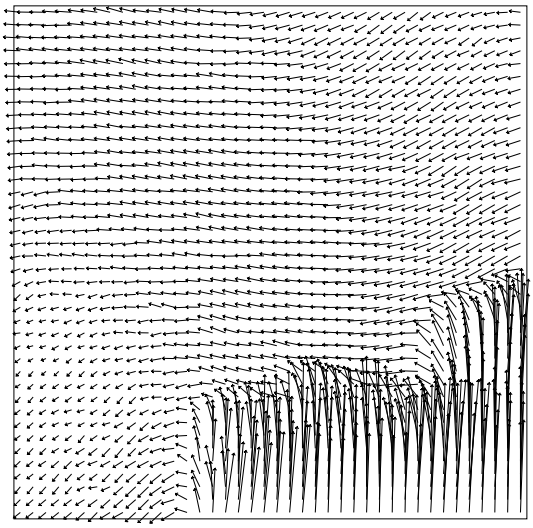
4 superimposed consecutive images



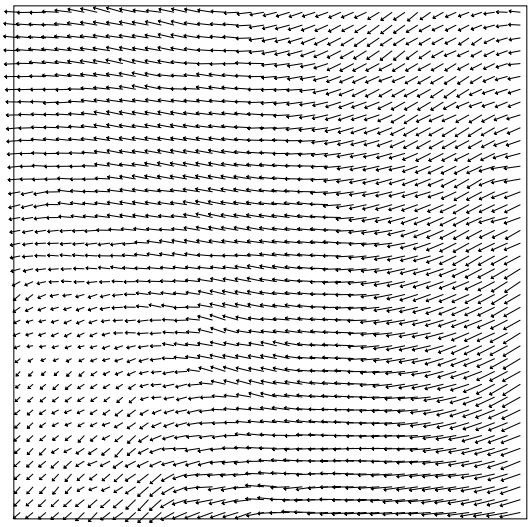
ODP, *X* and *Y* components



SODP, *X* and *Y* components



ODP, *X* and *Z* components



SODP, *X* and *Z* components

Fig. 4 Comparison of results obtained with ODP and SODP methods (STD331 sequence).