



HAL
open science

Vision-based correspondence using relaxation algorithms for particle tracking velocimetry.

Théo Benkovic, Jean-François Krawczynski, Philippe Druault

► **To cite this version:**

Théo Benkovic, Jean-François Krawczynski, Philippe Druault. Vision-based correspondence using relaxation algorithms for particle tracking velocimetry.. Measurement Science and Technology, In press, 10.1088/1361-6501/abb437 . hal-02947162

HAL Id: hal-02947162

<https://hal.sorbonne-universite.fr/hal-02947162v1>

Submitted on 23 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Vision-based correspondence using relaxation algorithms for particle tracking velocimetry

Théo Benkovic, Jean-François Krawczynski & Philippe Druault

Sorbonne Université, CNRS, Institut Jean Le Rond d'Alembert, F-75005 Paris, France

E-mail: jean-francois.krawczynski@sorbonne-universite.fr

16 August 2020

Abstract. A new particle tracking algorithm is derived based on consolidated methods with the aim to overcome the current limits encountered with high particle density flows. The proposed method consists in an integration of the relaxation algorithm based on match probabilities into vision-based features association concepts. Hybridization with PIV pre-analysis is suggested to help with parameters estimation. A dual calculation strategy is also developed in order to reduce the amount of spurious vectors. Simulation tests using synthetically generated images are carried out to evaluate the sensitivity of the proposed method to the particle image density, the background noise and the nature of the flow. Three flow configurations with a growing degree of complexity are successively considered: a 2-D flow over a moving-wall, a steady 2-D Lamb-Oseen vortex ring, and a 3-D unsteady homogeneous isotropic turbulence. The ability of the new tracking algorithm to provide better matching performances with high reliability than conventional techniques out of a dense particle image field is demonstrated.

Keywords: PTV, particle tracking, relaxation algorithm, vision-based technique, high particle density

Submitted to: *Meas. Sci. Technol.*

1. Introduction

Velocity measurements based on image analysis is one of those dynamic fields with about three decades of tremendous contributions which totally turned experimental flow motion characterization from flow-visualization to robust and efficient quantitative measurements. This is largely true as long as the flow to be characterized is without large changes in enthalpy, density or composition. For example, planar measurements allow the physical phenomena coupled with the medium and largest scales of the turbulent flows to be retrieved. As high- Re flows develop across a spectrum of scales of

considerable span, they still suffer from inherent limitations which put the small scales characteristics quantities of the turbulent flow beyond the reach of direct measurements.

The inherent averaging effect over the so-called interrogation window proscribes Particle Imaging Velocimetry (PIV) from measuring the maximum velocity gradients (Scarano 2003, Nogueira et al. 2005). Therefore, small-scale characteristic quantities of the turbulent flows, such as the mean energy dissipation rate, are still challenging to be determined from the direct differentiation of the data (Saarenrinne & Piirto 2000, Krawczynski *et al.* 2010). As for Particle Tracking Velocimetry (PTV), its spatial resolution mostly relies on the spacing between individual particles it tracks and the time interval, t between frames. Therefore, the spatial resolution increases *a priori* together with the particle image density, N_{ppp} defined as the number of particles per image area. Unfortunately, N_{ppp} is a major limiting factor for PTV reliability since large particle density prevents an optimized particle identification and localization: promoting large density results in particle images overlapping, i.e. indistinguishable patches of high-intensity pixel. This is formalized as the speckle phenomenon which is likely to occur when the source density, defined as the average number of imaged particles multiplied by the squared equivalent diameter of the particle image, exceeds one (Adrian 1984).

Should we see it as a consequence yet, most of the previous PTV dedicated studies did not (or failed to) consider the detection of individual particles in a high- N_{ppp} context. To the authors knowledge, most of the efforts were oriented towards the design of robust and efficient tracking algorithms at moderate N_{ppp} : very few studies dealt with $N_{ppp} > 0.06$ particles.pixel⁻¹. Brevis et al. (2011) proposed to combine the consolidated cross-correlation method (Yamamoto et al. 1993, Ishikawa et al. 2000) with an iterative relaxation algorithm based on match probabilities (Baek and Lee 1996, Ohmi & Li 2000, Pereira et al 2006). The proposed approach aimed at attenuating the limitations of both methods: the inability of the cross-correlation method to track features in the presence of strong velocity gradients due to the loss of similarity in the particles patterns, and the difficulty of the relaxation method to match particles embedded in low-information neighborhood. The integration of both methods was designed as a two-steps process: the cross-correlation method was the pre-processing stage of the relaxation method, and the weighting was controlled by the user-defined correlation threshold level. Particle matching based on vision principles was also proposed with the aim at dealing with “severely” overlapped particle image field (Lei et al. 2012). It relied on the principle of proximity, which favored short matches over long ones, and also on the principles of exclusion and similarity. The cross-correlation algorithm was added as an additional term in the so-called proximity matrix, and the weighting was in this case auto-adaptive. Displacement-shifted methods utilizing the guidance of PIV were designed as a modification of the proximity matrix in order to favor the particles that lie at the “expected location” (Paul et al. 2014).

This work subscribes in the earlier inquiry of our group specifically dedicated to the particle identification and localization issue under high-density particle conditions (Cheminet et al. 2018). The approach was based on the particle position reconstruction

through the inversion of a linear model connecting the PTV signal with a particle-based representation of the 3-D to 2-D projection. Much better detection performances with higher reliability than conventional techniques was demonstrated at particle image density as high as $N_{ppp} = 0.10$. We now propose to address the tracking issue under similar conditions.

The paper is organized as follows. The proposed tracking algorithm is mathematically detailed in section 2 and further details about its computational implementation are given in section 3. A description of the synthetic database is presented in section 4. The performances of the new algorithm are thoroughly compared to those of consolidated methods in section 5. Conclusions are made in section 6.

2. Algorithm

The proposed algorithm derives from the adaptation of the vision-based feature association concept (Scott and Longuet-Higgins 1991) to PTV (Lei et al. 2012), the relaxation method-based PTV (Baek and Lee 1996, Brevis et al. 2011) and the dual calculation method (Baek and Lee 1996, Jia et al. 2015). Its implementation is hybridized with PIV results in order to drastically reduce the use of arbitrary chosen parameters. In the following, a description of the new algorithm is given.

Let I and J be two short time-delayed frames with a grayscale intensity distribution resulting from the in-plane projection of illuminated particles. Let \mathbf{x}_i ($i = 1 \dots m$) and \mathbf{x}_j ($j = 1 \dots n$) be the locations of the particles in both frames which the PTV algorithm aims to put in a one-to-one correspondence. The algorithm consists of four steps.

Step 1. Build a *correlation-weighted proximity* matrix $\mathbf{G} \in \mathbb{M}_{m,n}$ of the two sets of particle locations. The underlying concept is intimately related to the visual correspondence principles outlined by Ullman (1979) : the *principle of similarity*, the *principle of proximity* and the *principle of exclusion* (only one-to-one matchings are allowed). Scott and Longuet-Higgins (1991) proposed a simple method to satisfy both the proximity and exclusion principles by exploiting some properties of the singular value decomposition (SVD) to a Gaussian-weighted distance matrix \mathbf{G}^d

$$G_{ij}^d = e^{-\frac{r_{ij}^2}{2\sigma^2}} \quad i = 1 \dots m, j = 1 \dots n \quad (1)$$

where $r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ is the Euclidean distance in the reference system of the frame between two sets of features, or points of interest, to be matched and σ is a user-defined parameter related to a characteristic distance: a small value of σ enforces short-distances pairings, while a large value permits large-scale displacements.

Pilu (1997) proposed to incorporate a normalized cross-correlation coefficient \mathbf{C} , further modified as a Gaussian weighted correlation coefficient by Lei et al. (2012), into the proximity matrix in order to embed the similarity principle:

$$G_{ij} = e^{-\frac{r_{ij}^2}{2\sigma^2}} \cdot e^{-\frac{(C_{ij}-1)^2}{2\gamma^2}} \quad i = 1 \dots m, j = 1 \dots n \quad (2)$$

where γ is a user-defined parameter to control the decay of the similarity weighting term. While in the original work of Scott and Longuet-Higgins (1991) and in later modifications (Pilu 1997, Lei et al. 2012) the singular value decomposition is directly performed to the proximity matrix, we here propose an alternative procedure in which \mathbf{G} is used as an initialization of the second step described hereafter.

Step 2. Perform an *iterative relaxation labelling technique* (Baek and Lee 1996). This method consists in computing a matching probability $\mathbf{P}_{ij} \in \mathbb{M}_{m,n}$ of a displacement $\underline{\mathbf{d}}_{ij}$ (therefore, of a correspondence between features) and its associated no-match probability $\mathbf{P}_i^* \in \mathbb{M}_m$ based on an iterative procedure which relies on the displacements of selected neighbours $\underline{\mathbf{d}}_{k(i)l(k)}$. By definition, a neighbour particle is a particle located within a radius T_m with respect to $\underline{\mathbf{x}}_i$ whereas the selected neighbours are those located at $\underline{\mathbf{x}}_{k(i)}$ that satisfy the quasi-rigidity condition $\|\underline{\mathbf{d}}_{ij} - \underline{\mathbf{d}}_{k(i)l(k)}\| < T_q$ (Baek and Lee 1996). The iterative probability update process was extensively presented in Baek and Lee (1996). It is briefly summarized in the following. The updated matching probability and no-match probability at the t^{th} iteration step are calculated as:

$$\mathbf{P}_{ij}^{(t)} = \frac{A\mathbf{P}_{ij}^{(t-1)} + B\mathbf{Q}_{ij}^{(t-1)}}{\mathbf{N}_{ij}^{(t-1)}} \quad (3)$$

$$\mathbf{P}_i^{*(t)} = \frac{\mathbf{P}_i^{*(t-1)}}{\mathbf{N}_{ij}^{(t-1)}} \quad (4)$$

where A and B are positive constants equal to 0.3 and 4 (Brevis et al. 2011) respectively, \mathbf{Q}_{ij} is the sum of the neighbours' probabilities

$$\mathbf{Q}_{ij}^{(t-1)} = \sum_{k(i)=1}^m \sum_{l(k)=1}^n \mathbf{P}_{k(i)l(k)}^{(t-1)}, \quad (5)$$

and \mathbf{N}_{ij} is a normalization term,

$$\mathbf{N}_{ij}^{(t-1)} = \sum_{j=1}^n \left(A\mathbf{P}_{ij}^{(t-1)} + B\mathbf{Q}_{ij}^{(t-1)} \right) + \mathbf{P}_i^{*(t-1)} \quad (6)$$

in order to enforce at every iteration step:

$$\sum_{j=1}^n \mathbf{P}_{ij} + \mathbf{P}_i^* = 1. \quad (7)$$

The interested reader is invited to refer to the works of Baek and Lee (1996) and Brevis et al. (2011) for more information on the relaxation method and its implementation.

Step 3. Determine a forward pairing matrix $\mathbf{M}^f \in \mathbb{M}_{m,n}$ that maximizes the inner product, $M^f : P = \text{trace}((M^f)^T P)$. Here, the superscript f refers to the forward computation, i.e. from the image I to the image J . We here go back to the approach proposed by Scott and Longuet-Higgins (1991) but instead of performing

the singular value decomposition of the proximity matrix (\mathbf{G}^d or \mathbf{G}), we propose to find the SVD of $\mathbf{P} \in \mathbb{M}_{m,n}$, such that

$$\mathbf{P} = \mathbf{T}\mathbf{D}\mathbf{U}^T \quad (8)$$

where $\mathbf{T} \in \mathbb{M}_m$ and $\mathbf{U} \in \mathbb{M}_n$ are orthogonal matrices and the diagonal matrix $\mathbf{D} \in \mathbb{M}_{m,n}$ contains the singular values in a descending numerical order along its diagonal elements. The matrix \mathbf{D} is then converted to a new matrix $\mathbf{E} \in \mathbb{M}_{m,n}$ by replacing every diagonal element D_{ii} with 1. The pairing matrix \mathbf{M}^f is finally obtained by computing

$$\mathbf{M}^f = \mathbf{T}\mathbf{E}\mathbf{U}^T \quad (9)$$

This final step enables the algorithm to embed the exclusion principle which arises from the orthogonality of the matrix \mathbf{M}^f . Mathematical proof of the optimized pairing process is given in Appendix A.

Step 4. Repeat Step 1., Step 2. and Step 3. in the backward direction, i.e. from the “second image” J to the “starting image” I . Therefore, a backward pairing matrix $\mathbf{M}^b \in \mathbb{M}_{n,m}$ is obtained (the upperscript b refers to the backward computation). The final pairing matrix $\mathbf{M} \in \mathbb{M}_{m,n}$ is computed as (Jia et al. 2015)

$$\mathbf{M} = \left(\mathbf{M}^f \cdot (\mathbf{M}^b)^T \right)^{1/2} \quad (10)$$

where (\cdot) is the Hadamard product operator.

This algorithm is referred to as Vision-Based Relaxation-Method, VBRM for brevity.

3. Outline of the methodology and Computational implementation

The whole process is outlined as follows:

- 1 Extract centers of the detected particles, $\underline{\mathbf{x}}_i$ ($i = 1 \dots m$) and $\underline{\mathbf{x}}_j$ ($j = 1 \dots n$) from two successive images of particles. This initial (and crucial) step which usually refers to the detection and localization of the particles in PTV is actually left aside from this study. Here, we choose instead to only focus on the matching/tracking step. Hence, the evaluation of the proposed algorithm is performed over synthetic images of known-locations particles.
- 2 Select candidates and identify neighbours. The term candidate particles refers to a set of particles located at the second frame that potentially are in one-to-one correspondence with a given particle in the first frame. They are generally found with some user-defined selection criteria, such as the expected maximum displacement. In the present work, the expected maximum local displacement is estimated after an initial guess of the displacement field through a PIV analysis. The term neighbour particles refers to a set of particles in the first frame located within a radius T_m with respect to $\underline{\mathbf{x}}_i$. T_m is an estimation of the size of the flow structures in which the particles exhibit similar motion. In this study, T_m is set to 15 pixels for all the test-cases discussed.

- 3 Build $\mathbf{I} \in \mathbb{M}_{m,L}$ and $\mathbf{J} \in \mathbb{M}_{n,L}$ of L -dimensional features. Features consist of images patches (i.e. pixels intensities) with L pixels centered around the particle positions. Therefore, row i (resp. j) of \mathbf{I} (resp. \mathbf{J}) is the row vectorization of a patch around the i th (resp. j th) feature-point of the first (resp. second) image. Moreover, every row of \mathbf{I} and \mathbf{J} is normalized to zero mean and unit norm. As a consequence, the inner product of \mathbf{I} and \mathbf{J} is nothing else than the correlation coefficient matrix in relation (2)

$$\mathbf{C} = \mathbf{J}\mathbf{I}^T \quad (11)$$

The computation of the correlation-weighted proximity matrix \mathbf{G} according to relation (2) is then trivial. The parameter γ is set to 0.12 for the present implementation and σ is a flow-dependent parameter estimated as a characteristic length scale of the displacement vectors field from the PIV analysis (Lei et al. 2012). The rows of \mathbf{G} are then set to zero, except for the j th column corresponding to a candidate of the i th particle. The resulting \mathbf{G} matrix is sparse. It is to be noticed that no other filtering procedure is performed up to this stage.

- 4 Initialization of the iterative relaxation method. In an attempt to integrate the classical cross-correlation (CC) and relaxation (RM) algorithms, Brevis et al. (2011) proposed to convert to initial probabilities the correlation levels calculated during their initial CC stage. Here, we choose instead to use the information embedded in \mathbf{G} as,

$$\mathbf{P}_{ij}^{(0)} = \frac{\mathbf{G}_{ij}}{\sum_{j=1}^n (\mathbf{G}_{ij}) + (1 - \max_{j=1\dots n} \mathbf{G}_{ij})} \quad (12)$$

$$\mathbf{P}_i^{*(0)} = \frac{1 - \max_{j=1\dots n} \mathbf{G}_{ij}}{\sum_{j=1}^n (\mathbf{G}_{ij}) + (1 - \max_{j=1\dots n} \mathbf{G}_{ij})}. \quad (13)$$

In the work of Brevis et al. (2011) the matrix \mathbf{G} in relations (12) and (13) was replaced by an equivalent form of matrix \mathbf{C} and a matching probability equal to 1 was set for the particles matched by their CC step.

- 5 Iterate with the probability update according to relations (3) and (4). The iterative process is stopped when the difference between the probabilities in the previous and current iterations is less than a certain threshold λ . In the present implementation, a value of $\lambda = 0.01$ is used.
- 6 Compute the forward pairing matrix \mathbf{M}^f according to relations (8) and (9).
- 7 Eventually, in a similar manner as Jia et al. (2015) a *dual calculation method* is applied, i.e. steps 2-6 are repeated by considering the backward tracking from the “second image” to the “starting image” to compute the backward pairing matrix \mathbf{M}^b . The final pairing matrix, denoted \mathbf{M} for simplicity is computed according to relation (10).

The code of our tracking algorithm, denoted VBRM in the rest of the paper, is implemented in Matlab and makes use of its high vectorization level and its optimized built-in functions.

4. Synthetic set-up

The matching capabilities of the proposed algorithm are evaluated against well-known particle locations obtained from three characteristic flow simulations. The interest of using synthetic images is twofold: (a) the exact position of the particles is known; (b) a direct comparison between the matches given by the proposed algorithm and those by other referenced methods is possible. For every flow simulation, double-exposure snapshots of $N_x \times N_y = 512 \times 512$ pixel at 8-bit grayscale are synthesized. While the particles' locations of the first frames are systematically randomly distributed within the entire image, their time-delayed positions within the second frame are computed with the velocity field obtained from the flow simulations. Details of the particle images generation together with the simulated flow characteristics are given in the following.

4.1. Particle images generation

The full procedure detailed in Cheminet et al. (2018) is used to synthesize the images of particles. It is briefly summarized as follows. The image formation of the projected seeded particles in the illuminated volume onto the image plane is monitored by the so-called point spread function (PSF) modeling the formation of the Airy-spots. The PSF is written as a 2-D integrated Gaussian intensity distribution characterized by its standard deviation σ_{psf} . Considering monodisperse particles of diameter $d_p = 1\mu\text{m}$ within the illuminated volume and a camera of focal length $f = 200$ mm placed at 1 m orthogonally to the laser sheet plane, we choose to set $\sigma_{\text{psf}} = 0.6$ in order to obtain an averaged particles diameter of 2.4 pixels (except for the first test-case (i) presented in section 4.2 where the particle diameter is fixed to 4 pixels). The maximum intensity of the 8-bit grayscale images is assumed to be an exponentially decreasing function of the particles in-depth position in the laser sheet (set to 1 mm) only. Let us notice that the particles are systematically generated on a larger domain than the frames dimension in order to control the particle seeding density, N_{ppp} despite the inter-frame excursions of the particles due to the flow of interest.

To mimic experimental measurement noise conditions and to evaluate their influence on the tracking method efficiency, dedicated series of particle images are generated. To this end, a Gaussian noise with zero-mean and a standard deviation of 5 % is added to each image of specific series.

Some of the relevant parameters describing the synthetic images are summarized in table 1.

4.2. Simulated flows

Three flow configurations with a growing degree of complexity are successively considered: (i) a 2-D flow over a moving-wall, (ii) a steady 2-D Lamb-Oseen vortex ring, (iii) a 3-D unsteady homogeneous isotropic turbulence (HIT).

- (i) *2-D flow over a moving-wall*. We intentionally reproduce a referenced test case (Lei et al. 2012). The synthetic time-delayed images are generated according to the analytical velocity profile derived from the Stokes' first problem,

$$u = U \times (1 - \operatorname{erf}(y/2\sqrt{\mu t})) \quad (14)$$

with $U = 10$, $\mu = 5$, and $t = 75$. The particle seeding density N_{ppp} is fixed to 0.03 particles.pixel⁻¹ and the particle diameters are equal to 4 pixels.

- (ii) *2-D steady Lamb-Oseen (L0) vortex ring*. This test case relies on an exact solution of the two-dimensional Navier Stokes equations. The simulated flow consists of two steady counter-rotating vortices. Each of them is described as follows in a uniform cartesian meshgrid:

$$u_\theta = u_{\theta_{max}} \left(1 + \frac{1}{2\alpha} \right) \frac{r_0}{r} \left(1 - \exp \left(1 - \alpha \left(\frac{r}{r_0} \right) \right) \right) \quad (15)$$

where u_θ is the angular velocity component, r is the distance from the vortex center, r_0 (set to 1/5 of the image width) is the core radius defined as the distance where the maximum angular velocity $u_{\theta_{max}} = 5 \text{ pixel.s}^{-1}$ is reached and $\alpha = 1.26$ (Devenport et al. 1996). The Oseen vortices are symmetrically placed at $(N_x/2, +N_y/5)$ and $(N_x/2, -N_y/5)$ within the image, respectively. As an illustration, figure 1 (left) displays the steady velocity vectors field superimposed onto the velocity modulus.

- (iii) *HIT flow*. The homogeneous and isotropic Johns Hopkins Turbulence Database (JHTDB) is used to design a more realistic test case (Li et al. 2008). This 3-D database was recently used for the validation of tracking algorithms in the context of 3-D PTV (Clark et al. 2018, Kozul *et al.* 2019), yet never to our knowledge in a 2-D PTV context. The DNS was performed on a spatial domain of $2\pi \times 2\pi \times 2\pi$ corresponding to a regular mesh of 1024^3 grid points. For the present investigation, an instantaneous 3-D velocity field is extracted from a sample volume \mathcal{V} , with a regular mesh of $576 \times 576 \times 7$ grid points along the (x, y, z) directions respectively. The z -symmetric plane of \mathcal{V} is supposed to be illuminated with a laser sheet of Gaussian intensity profile characterized by a standard deviation of 5%. Note that for this particular test case, the synthetic image generation accounts for the in-plane Gaussian intensity profile of the illumination, i.e. that is brighter in the center of the image and dimmer at the left and right edges. Because of the intrinsic three-dimensionality of the flow, the entire \mathcal{V} volume is initially randomly seeded with particles. Advection of each particle is performed within \mathcal{V} by the instantaneous velocity field interpolated from the DNS database to account for the inter-frames excursions in the transverse direction z .

The resulted 2-D measurement plane is discretized on a 512×512 pixel meshgrid. As an illustration, figure 1 (right) displays an instantaneous velocity vectors field superimposed onto the velocity modulus.

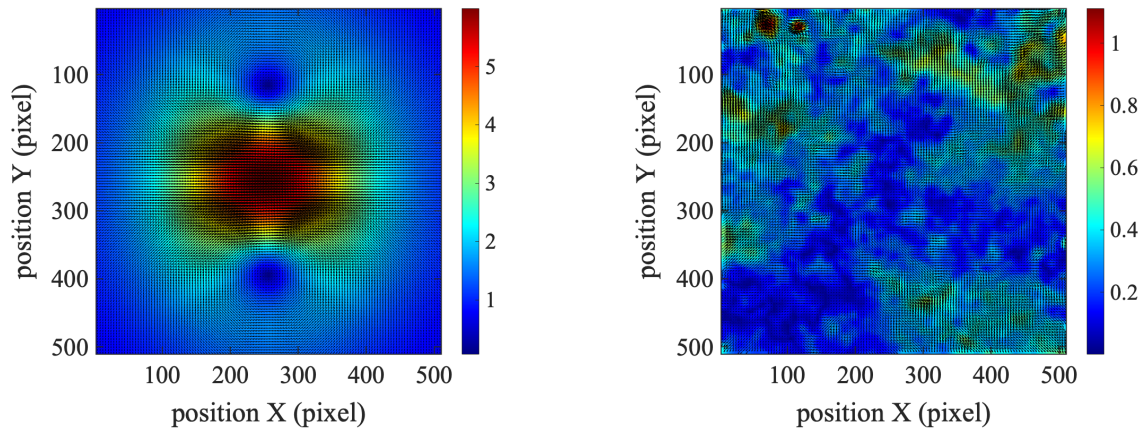


Figure 1: Left: 2-D Lamb-Oseen vortex ring. Velocity vectors field superimposed onto the velocity modulus. Right: HIT test case. Instantaneous fluctuating velocity vectors field superimposed onto the velocity modulus. Velocity modulus is color-coded in pixel/s units.

Table 1: Particle images parameters.

Flow description	moving wall	LO vortex ring	HIT
Image size (pixel \times pixel)	512 \times 512	512 \times 512	512 \times 512
# pairs of images	20	20	20
Mean d_p (pixel)	4	2.4	2.4
Min N_{ppp} (particles.pixel $^{-1}$)	0.03	0.02	0.10
Max N_{ppp} (particles.pixel $^{-1}$)	0.03	0.12	0.10
Min # particle	7711	4937	25850
Max # particle	8135	31482	26410
Min IFPC ratio (%)	99.85	99.77	81.98
Max IFPC ratio (%)	99.99	99.97	83.08
Min N_{ppp}^* (particles.pixel $^{-1}$)	0.0298	0.0197	0.081
Max N_{ppp}^* (particles.pixel $^{-1}$)	0.0298	0.1195	0.083

d_p is the particle image diameter, IFPC is the inter-frame particles correspondence ratio,

N_{ppp}^* is particle density corrected for the inter-frame excursions.

4.3. Performance metrics

The performance of our tracking method is quantified and compared with those obtained by other referenced methods by computing two characteristic metrics initially introduced in the context of point tracking methods (Yilmaz et al. 2006):

$$\text{recall} = \frac{\# \text{ of correct correspondences}}{\# \text{ of actual correspondences}} \quad (16)$$

10

$$\text{precision} = \frac{\# \text{ of correct correspondences}}{\# \text{ of established correspondences}} \quad (17)$$

where actual correspondences refer to the correspondences available in the ground truth. Indeed, as stated above, the actual amount of particle match depends not only on the total amount of particles in two successive frames (therefore to the seeding density) but also on the flow configuration. When 3-D flows are considered as in the HIT test case, out-of-plane excursions of the particles are inherent to the flow. They consequently leave us with fewer pairs of particles than particles themselves. We thus defined the particle density corrected for the inter-frame excursions, $N_{ppp}^* = \text{IFPC} \times N_{ppp}$ as the number of particles per image area that are effectively in one-to-one correspondence between two successive frames, see table 1. For 2-D flow cases, the slightly reduced N_{ppp}^* compared to N_{ppp} accounts for the particles excursions accross the left-right and top-bottom boundaries of the images. As for 3-D flows, the out-of-plane excursions of the particles possibly account for the largest contribution to the decrease in IFPC ratio. For the present investigation, the mean value of the IFPC ratio (computed from 20 pairs of images) is set to 82.5 % by an appropriate choice of the time-step between two successive frames.

A correspondence is considered as correct if it satisfies simultaneously two conditions:

$$\frac{1}{\pi} \cos^{-1} \frac{\mathbf{v}_{\text{meas}} \cdot \mathbf{v}_{\text{ref}}}{\|\mathbf{v}_{\text{meas}}\| \cdot \|\mathbf{v}_{\text{ref}}\|} < 0.05 \quad (18)$$

$$\frac{\|\mathbf{v}_{\text{meas}} - \mathbf{v}_{\text{ref}}\|}{\|\mathbf{v}_{\text{ref}}\|} < 0.05 \quad (19)$$

where \mathbf{v}_{meas} is the velocity vector obtained with a tracking algorithm and \mathbf{v}_{ref} is the referenced vector at the same location obtained either analytically, test-cases (i) and (ii), or by interpolation, test-case (iii). Most of the previous studies only considered a condition based on the norm evaluation (19), furthermore with a weaker tolerance of either 1- or 0,5- pixel than (19). The directional condition (18) was rarely evaluated. We thus consider that the proposed quantification of the correct matches enables a finer discrimination between the tested tracking methods.

5. Results

Let us now attempt to assess the robustness and the efficiency of our tracking method within representative experimental conditions. To this end, numerical tests are performed on synthetic particle images for three flow configurations. Apart the inherent influence of the flow towards the quality of the results, the effect of particle density, N_{ppp} and the effect of the background noise in the particle images is also evaluated. Most of the forward analysis relies on the evaluation of the performance metrics introduced in section 4.3. To ensure statistical convergence of the results discussed bellow, sets of 20 pairs of synthetic images are generated for every N_{ppp} , see table 1. The performances of

the VBRM method introduced in this study are compared to those of the consolidated ICCRM (Brevis et al. 2011) and VB-PTV (Lei et al. 2012) methods. The PTVlab software (Brevis et al. 2011) is used as a Matlab implementation of the ICCRM tracking algorithm. Our tracking algorithm and the tracking part of VB-PTV, as recommended in Lei et al. (2012), are guided by PIV results. The open source PIV algorithm (Thielicke and Stamhuis 2014) is used to compute initial guess of the velocity fields and to help with user-defined parameters estimation, as noted in section 3. An outlier removal algorithm (Duncan et al. 2010) is applied to all the PTV data.

5.1. Case (i): Flow over a moving wall.

Table 2 shows the performance metrics values as obtained by the three tested algorithms from this Stokes' first problem test case. All the matching algorithms are capable of excellent accuracy. Although our 5% over the norms criteria for the metrics evaluation (section 4.3) are, over the whole velocity profile, always more severe than the 0.5 - pixel maximal tolerance retained in Lei et al. (2012), similar global mean performances are evidenced on this simple configuration between the three tracking methods and those reported in Lei et al. (2012). A further insight is given to the results of these simulations with the (local) representation of the velocity profile as computed from a pair of moving wall 512×512 pixel images by ICCRM, VB-PTV and VBRM, see figure 2. This illustrates the capability of our tracking method to generate smaller dispersions than referenced algorithms with respect to the analytical profile. Similar behaviours are obtained independantly of the considered couple of images (not shown here).

Table 2: Performance metrics of each tracking algorithm from the moving-wall flow case.

Algorithm	recall	precision
ICCRM (Brevis et al. 2011)	99.69	99.83
VB-PTV (Lei et al. 2012)	97.98	99.15
VBRM (present work)	99.05	99.93

5.2. Case (ii): 2-D Lamb-Oseen vortex ring

We aimed into gaining some insight into our tracking methods performance in a high-density of neighborhood particles and into the impact that measurement noise can have on PTV results. Particle image density, N_{ppp} is a major limiting factor for PTV reliability since large density inhibits an optimized particle identification and localization. Severely overlapped particles which signal is discretized on a coarse meshgrid also result in non-distinguishable intensity patches for the classical cross-correlation algorithm. As for the background noise, it is generally associated with non-uniform variations of grey levels in the image not related to the particle. Hence

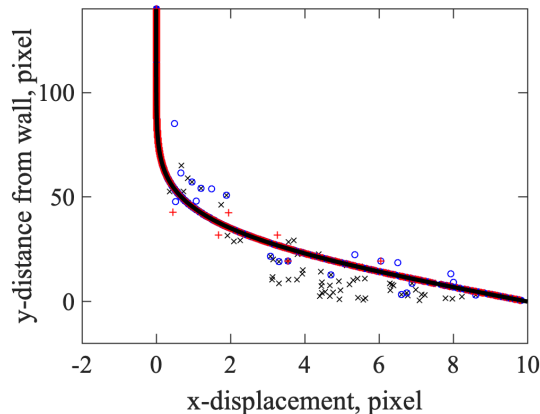


Figure 2: Velocity profile of the moving-wall flow. Scattered PTV data deduced from each algorithm, VB-PTV (black \times), ICCRM (blue \circ) and VBRM (red $+$), superimposed onto the analytical profile (solid black line).

it alters the ability of any tracking method based on constant intensity illuminated patterns to accurately match particles, even though they remain in the same plane of constant illumination. Synthetic images without additional noise are first considered and the performances of each tracking algorithm against the influence of the particle density are discussed in section 5.2.1. The impact of the background noise on the same performance criteria is then assessed in section 5.2.2.

5.2.1. Effect of N_{ppp}

We now turn our analysis towards the sensitivity of the VBRM method to the particle density, N_{ppp} . The mean performance metrics are presented in figure 3 as a function of the particle density, N_{ppp} for the three tracking methods. Averaging is performed over every N_{ppp} . In the simulations of this test, the particle image density is varied from $N_{ppp} \approx 0.02$ to $N_{ppp} \approx 0.12$.

The number of valid matches as recovered by VBRM remains at a very high level (actually very close to the exact amount of simulated correspondences), even at large particle density. This statement is emphasized by the recall values which are quasi constant and very close to 1 for all the investigated N_{ppp} . The capability of our method to recover a large amount of valid correspondences is further illustrated by the precision coefficient which also remains very close to 1, for all the investigated particle densities. This result shows that, not only VBRM is efficient as it “finds” a quasi-maximal amount of correspondences, but also accurate as it is not prone to produce large amount of outliers. Therefore, the improvement of the particles correspondences with our method is evidenced when compared to referenced algorithms in the same conditions. Indeed, ICCRM and VB-PTV fail at recovering a large amount of matches when N_{ppp} is increased as illustrated by the clear drop of their recall ratio.

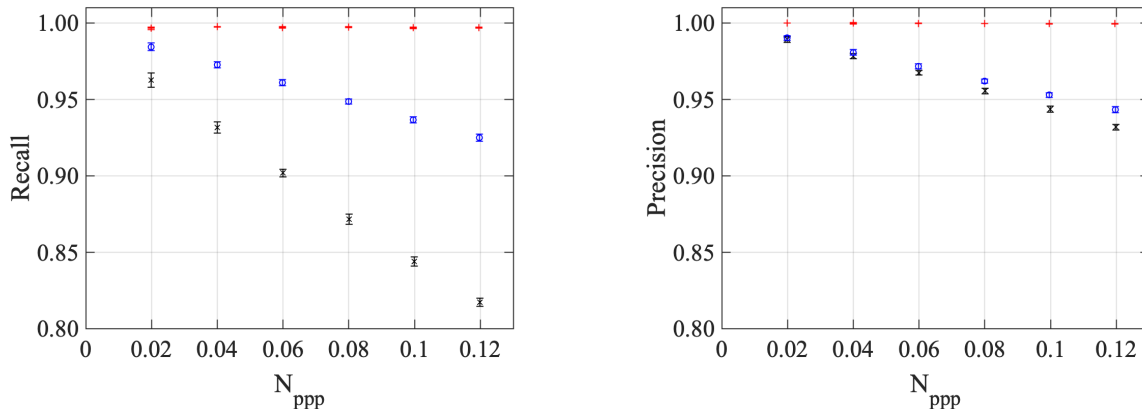


Figure 3: Quality of the tracking as given by VB-PTV (black \times), ICCRM (blue \circ) and VBRM (red $+$) as a function of the particle density, N_{ppp} . Synthetic images without additive noise. Left: recall ; Right: precision.

5.2.2. Effect of background image noise

The high efficiency of VBRM is further emphasized with an insight on its sensitivity to the presence of an additive background noise. The mean performance metrics are presented in figure 4 as a function of the particle density, N_{ppp} for the three tracking methods.

Whatever the tracking method, only a slight decrease of the performance metrics is observed respectively to the previous test case, section 5.2.1. The above statements are therefore confirmed by the results of this simulation test: over the investigated N_{ppp} range, the recall and precision values obtained with VBRM remain very close to 1. For comparison, at $N_{ppp} \approx 0.12$, the recall mean values are 99.70%, 92.51% and 81.73% and the precision mean values are 99.96%, 94.34% and 93.19% for VBRM, ICCRM and VB-PTV respectively. Although good performances are obtained with the referenced ICCRM and VB-PTV methods, the improvement of the particles-matching procedure with our method is again clearly evidenced, particularly within the medium-to-large N_{ppp} range.

An illustration of the velocity-vectors field computed by each of the three algorithms is presented in figure 5 for a pair of images at $N_{ppp} = 0.12$. From a qualitative perspective, the reconstruction of the velocity-vectors field is satisfactory for such a 2-D flow test-case. However, the reconstructions as given by the ICCRM and VB-PTV algorithms exhibit more visible outliers than VBRM in a reduced number of locations.

To further refine this qualitative analysis, a zoom-in view of the same velocity-vectors field is depicted in figure 6 aside with the corresponding synthetic first-frame on which the particle centroids (of the two successive frames) are superimposed. Such a representation emphasizes that ICCRM and VB-PTV methods fail to retrieve one-to-one correspondences when multiple particle centroids are close all together, and also when the particle centroid is located at a low-intensity pixel. The superiority of VBRM

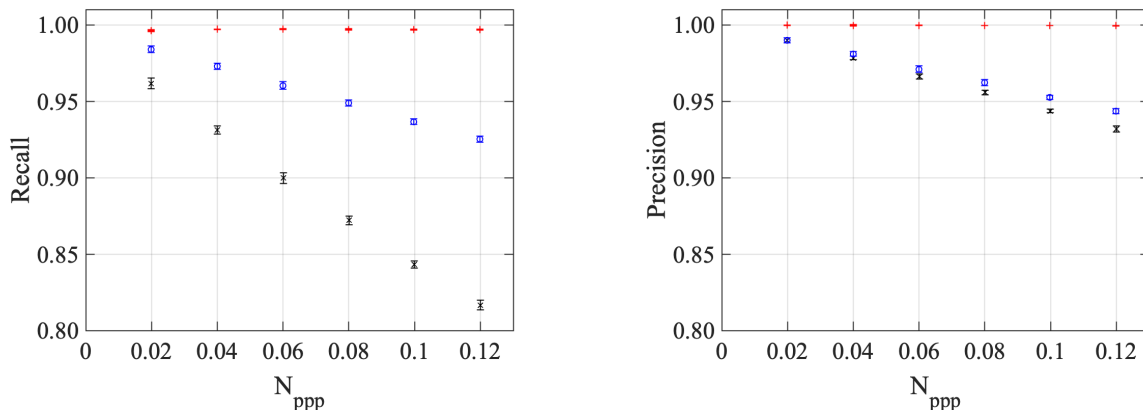


Figure 4: Quality of the tracking as given by VB-PTV (black \times), ICCRM (blue \circ) and VBRM (red $+$) as a function of the particle density, N_{ppp} . Noisy synthetic images. Left: recall ; Right: precision.

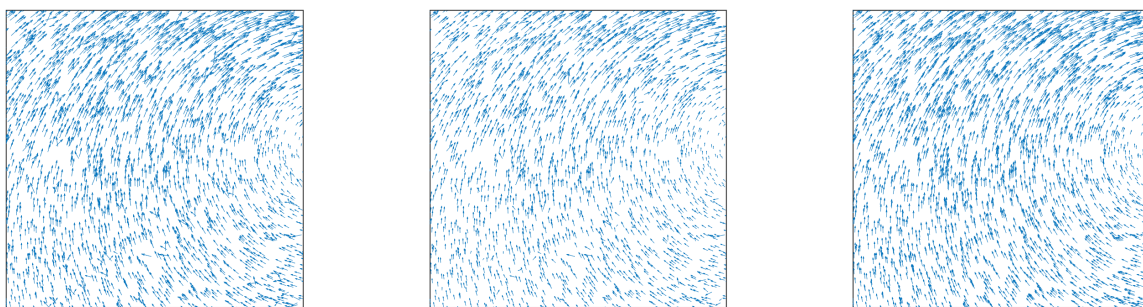


Figure 5: Zoom-in view (140×140 pixel) of a reconstructed velocity-vectors field from a pair of noisy image at $N_{ppp} = 0.12$. From Left to Right: ICCRM, VB-PTV and VBRM.

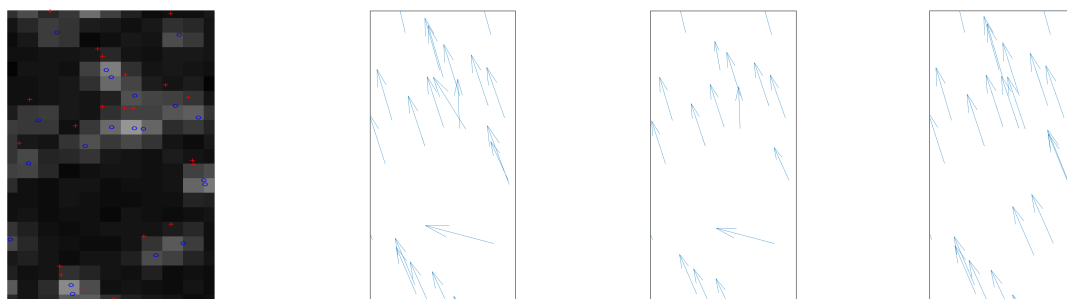


Figure 6: Qualitative comparison of reconstructed vector fields obtained by the three tested methods. Zoom-in views of 10×20 pixel. From Left to Right: particles centroids (blue \circ : first image ; red $+$: second “time-delayed” image) superimposed on a particle image ; ICCRM, VB-PTV and VBRM velocity vectors field reconstruction.

is particularly observed when two particle centroids are located at the same pixel.

5.3. Case (iii): 3-D unsteady HIT flow

The results presented in item (ii) have demonstrated the VBRM algorithm ability to efficiently track particles over a wide range of realistic parameters. So far, the simulation tests were carried out on particles seeded in two-dimensional flows. Meanwhile, the implementation of this new tracking method in realistic problems demands an evaluation of its ability to reconstruct velocity vectors field of three-dimensional flows. The non-negligible out-of-plane excursions of particles between two consecutive frames are related to a reduction of the ratio of the possible amount of correspondences to the total amount of particles, as indicated by the IFPC parameter in table 1. Therefore, the solitary particles can be seen as a non-additive noise contribution, which impact on the tracking performances must be evaluated. Let us remark that the time step between successive frames is adjusted in order to set the maximum amount of out-of-plane particles to 20%, i.e. an IFPC parameter no smaller than 0.8. Therefore, for this specific test case based on synthetic particle images at $N_{ppp} = 0.1$, the true particle-density corresponding to the amount of particles that can effectively be put in one-to-one correspondence is $N_{ppp}^* = 0.082$.

A comparison of the performance of our tracking method with those of the referenced ICCRM and VB-PTV is presented in table 3. The high efficiency and the superiority of the novel VBRM tracking algorithm is clearly indicated. VBRM provides recall and precision values both equal to 94.5%, which means that our algorithm permits a robust and an accurate velocity-vectors estimation in the measurement plane, despite the inherent complexity of the three-dimensional nature of the flow.

Both the reference velocity field interpolated from the available DNS database and the resulted vector field computed by VBRM are represented as a full-size illustration in figure 7. Their superimposition on a zoom-in view further emphasizes the great efficiency and accuracy of the VBRM algorithm.

Table 3: Performance metrics of each tracking algorithm from THI-3D database.

Algorithm	N_{ppp}^*	N_{ppp}	recall	precision
ICCRM (Brevis et al. 2011)	0.082	0.102	82.40	72.01
VB-PTV (Lei et al. 2012)	0.082	0.102	76.30	73.62
VBRM (present work)	0.082	0.102	94.52	94.45

6. Conclusion

A novel algorithm for particle matching in PTV was proposed. The approach presented derived from vision-based features association concepts with their underlying principles of similarity, proximity and exclusion. The principle of proximity and similarity were classically embedded in the construction of a proximity matrix between features to

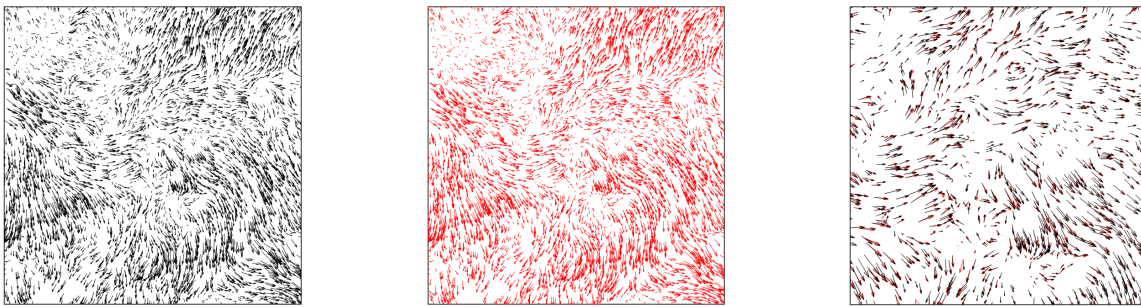


Figure 7: Illustration of the VBRM velocity vectors reconstruction. Left: (512×512 pixel image) Reference velocity field (interpolated from the DNS database) at the tracked particle position. Centre: (512×512 pixel image) VBRM Reconstructed velocity field. Right: zoomed view, 250×250 pixel area) Superimposition of the reference and the VBRM reconstructed velocity fields.

be put in one-to-one correspondence. The novelty of this approach was to use this proximity matrix as an initialization of an iterative relaxation labelling technique based on matching probabilities. The singular value decomposition usually performed on the proximity matrix was instead applied to the matching probability matrix accounting for the similarity of the local flow structure in this study. The principle of exclusion embedded in this former step was further guaranteed by the dual calculation approach implemented in our algorithm. Hybridization with PIV pre-analysis was also suggested to help with parameters estimation.

Performance tests were conducted with synthetically generated particle images and against an exploration of the generating conditions (particle density, background noise, 2-D or 3-D flow configuration).

Overall, the proposed technique was robust and efficient yet in noisy conditions, and the comparison with other conventional tracking methods demonstrated its ability to reliably match significantly more particles out of a dense particle image field ($N_{ppp} > 0.08$) and/or with large particle intensity pixels variation.

References

- Adrian R J 1984 Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow: speckle velocimetry vs particle image velocimetry. *Appl. Opt.* **23**(11), 1690-1691.
- Baek S.J. & Lee S.J. 1996 A new two-frame particle tracking algorithm using match probability. *Exp. Fluids* **22**(1), 23-32.
- Barnard S.T. & Thompson W.B. 1980 Disparity analysis of images. *IEEE Trans Pattern Anal Mach Intell.* (4), 333-340.
- Brevis W., Nino Y. & Jirka G.H. 2011 Integrating cross-correlation and relaxation algorithms for Particle Tracking Velocimetry. *Exp. Fluids* **50**(1), 135-147.
- Cheminet A., Krawczynski J.F. & Druault Ph. 2018 Particle image reconstruction for particle detection in particle tracking velocimetry. *Meas. Sci. Technol.* **29**(12), 125202.
- Clark A., Machicoane N. & Aliseda A. 2019 A quantitative study of track initialization of the four-frame

- best estimate algorithm for three-dimensional Lagrangian particle tracking. *Meas. Sci. Technol.* **30**(4), 045302.
- Devenport W.J., Rife M.C., Liapis S.I. & Follin G.J. 1996 The structure and development of a wing-tip vortex. *J. Fluid Mech.* **312**, 67-106.
- Duncan J., Dabiri D., Hove J. & Gharib M. 2010 Universal outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data. *Meas. Sci. Technol.* **21**(5), 057002.
- Ishikawa M, Murai Y, Wada A, Iguchi M, Okamoto K & Yamamoto F 2000 A novel algorithm for particle tracking velocimetry using the velocity gradient tensor. *Exp. Fluids* **29**(6), 519-531.
- Jia P., Wang Y., Zhang Y. & Yang B. 2015 Relaxation algorithm based PTV with dual calculation method and its application in addressing particle saltation. *J. Visualization* **18**(1), 71-81.
- Kozul M., Koothur V., Worth N.A. & Dawson J.R. 2019 A scanning particle tracking velocimetry technique for high Reynolds number turbulent flows. *Exp. Fluids* **60**(8), 137.
- Krawczynski J F, Renou B & Danaila L 2010 The structure of the velocity field in a confined flow driven by an array of opposed jets. *Physics of Fluids* **22**(4), 045104.
- Lei Y.C., Tien W.H., Duncan J., Paul M., Ponchaut N., Mouton C., Dabiri D., Rosgen T. & Hove J. 2012 A vision-based hybrid particle tracking velocimetry (PTV) technique using a modified cascade correlation peak-finding method. *Exp. Fluids* **53**(5), 1251-1268.
- Li Y., Perlman E., Wan M., Yang Y., Burns R., Meneveau C., Burns R., Chen S., Szalay A. & Eyink G. 2008 A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *J. Turb.* **9**, N31.
- Nogueira J., Lecuona A. & Rodriguez P.A. 2005 Limits on the resolution of correlation PIV iterative methods. Fundamentals. *Exp. Fluids* **39**(2), 305-313.
- Ohmi K & Li H.Y. 2000 Particle-tracking velocimetry with new algorithms. *Meas. Sci. Technol.* **11**(6), 603-616.
- Paul M., Tien W.-H. & Dabiri D. 2014 A displacement-shifted vision-based hybrid particle tracking velocimetry (PTV) technique. *Exp. Fluids* **55**(3), 1676.
- Pereira F., Stuer H., Graff E. & Gharib M. 2006 Two-frame 3D particle tracking. *Meas. Sci. Technol.* **17**(7), 1680.
- Pilu M. 1997 A direct method for stereo correspondence based on singular value decomposition. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 261-266.
- Saarenrinne P. & Piirto M. 2000 Turbulent kinetic energy dissipation rate estimate from PIV velocity vector fields. *Exp. Fluids* **29**(1), 300-307.
- Scarano F. 2003 Theory of non-isotropic spatial resolution in PIV. *Exp. Fluids* **35**(3), 268-277.
- Scott G. & Longuet-Higgins H. 1991 An algorithm for associating the features of two images. *Biol. Sci.* **244**(1309), 21-26.
- Schönemann, P. H. 1966 A generalized solution of the orthogonal procrustes problem. *Psychometrika* **31**(1), 1-10.
- Thielicke W. & Stamhuis E.J. 2014 PIVlab Towards User-friendly, Affordable and Accurate Digital Particle Image Velocimetry in MATLAB. *Journal of Open Research Software* **2**(1), 355-364.
- Ullman S. 1979 The interpretation of visual motion. *MIT Press, Cambridge, MA*.
- Yamamoto F., Uemura T., Tian H.Z. & Ohmi K. 1993 Three-Dimensional PTV Based on Binary Cross-Correlation Method. *JSME International Journal Series B* **36**(2), 279-284.
- Yilmaz A., Javed O. & Shah M. 2006 Object Tracking: A Survey. *ACM Comput. Surv.* **38**(4), 13.

Appendix A. Determination of the nearest matrix to an orthogonal matrix

This demonstration is adapted from Schonemann, P. H. (1966) to this specific vision-based problem.

Statement problem

Let $\mathbf{G} \in \mathbb{M}_{m,n}$ denote an orthogonal real matrix, the nearest matrix (in a statistical sense) of \mathbf{G} is an orthogonal matrix, $\mathbf{A} \in \mathbb{M}_{m,n}$ satisfying $\mathbf{A} = \mathbf{T}\mathbf{E}\mathbf{U}^T$ (T indicates the transpose matrix).

$\mathbf{E} \in \mathbb{M}_{m,n}$ is a matrix of same size as \mathbf{G} satisfying $E_{ij} = 1$ if $i = j$ and $E_{ij} = 0$ otherwise.

$\mathbf{T} \in \mathbb{M}_{m,m}$, $\mathbf{U} \in \mathbb{M}_{n,n}$ are orthogonal matrices. The columns of \mathbf{T} and \mathbf{U} are the orthogonal eigenfunctions of $\mathbf{G}\mathbf{G}^T$ and $\mathbf{G}^T\mathbf{G}$ respectively. Therefore $\mathbf{G} = \mathbf{T}\mathbf{D}\mathbf{U}^T$ with \mathbf{D} a rectangular matrix of same size as \mathbf{G} and with diagonal elements which are eigenvalues of \mathbf{G} .

Proofs

Mathematically, one has to minimize $tr(\mathbf{E}^T\mathbf{E})$ with the following constraints: $\mathbf{E} = \mathbf{A} - \mathbf{G}$ and $\mathbf{A}^T\mathbf{A} = \mathbf{I}$.

Let $g_1 = tr(\mathbf{E}^T\mathbf{E})$. Then $g_1 = tr((\mathbf{A} - \mathbf{G})^T(\mathbf{A} - \mathbf{G})) = tr(\mathbf{A}^T\mathbf{A} - 2\mathbf{A}^T\mathbf{G} + \mathbf{G}^T\mathbf{G})$. As $\mathbf{A}^T\mathbf{A} - \mathbf{I} = 0$, whatever $\mathbf{L} \in \mathbb{R}_{m,n}$, $\mathbf{L}(\mathbf{A}^T\mathbf{A} - \mathbf{I}) = 0$.

Let $g_2 = tr(\mathbf{L}(\mathbf{A}^T\mathbf{A} - \mathbf{I}))$ and $g = g_1 + g_2$. The real function $g(\mathbf{A})$ has an extremum if $\frac{\partial g}{\partial \mathbf{A}} = 0$. After calculation, one obtains

$$\frac{\partial g}{\partial \mathbf{A}} = 2\mathbf{A} - 2\mathbf{G} + \mathbf{A}(\mathbf{L} + \mathbf{L}^T) \quad (\text{A.1})$$

This implies that an extremum is reached if $\mathbf{G} = \mathbf{A} + \mathbf{A}\mathbf{Q}$ with $\mathbf{Q} = (\mathbf{L} + \mathbf{L}^T)/2$. Immediately, \mathbf{Q} is a symmetric matrix $\mathbf{Q} = \mathbf{A}^{-1}\mathbf{G} - \mathbf{I} = \mathbf{Q}^T$ with $\mathbf{A}^{-1} = \mathbf{A}^T$ (orthogonal matrix). Then one obtains: $\mathbf{A}^T\mathbf{G} = (\mathbf{A}^T\mathbf{G})^T = \mathbf{G}^T\mathbf{A}$ leading to

$$\mathbf{G}^T\mathbf{G} = \mathbf{A}^T\mathbf{G}\mathbf{G}^T\mathbf{A} \text{ or } \mathbf{G}\mathbf{G}^T = \mathbf{A}\mathbf{G}^T\mathbf{G}\mathbf{A}^T \quad (\text{A.2})$$

The diagonalization of both symmetric $\mathbf{G}\mathbf{G}^T$ and $\mathbf{G}^T\mathbf{G}$ matrices having the same eigenvalues, leads to $\mathbf{G}\mathbf{G}^T = \mathbf{T}\mathbf{D}_p\mathbf{T}^T$ and $\mathbf{G}^T\mathbf{G} = \mathbf{U}\mathbf{D}_p\mathbf{U}^T$ where \mathbf{T} and \mathbf{U} are orthogonal matrices and the diagonal entries of \mathbf{D}_p are the eigenvalues of both matrices. Following equation A.2, $\mathbf{T}\mathbf{D}_p\mathbf{T}^T = \mathbf{A}\mathbf{U}\mathbf{D}_p\mathbf{U}^T\mathbf{A}^T$ that means $\mathbf{T} = \mathbf{A}\mathbf{U}$ and $\mathbf{A} = \mathbf{T}\mathbf{E}\mathbf{U}^T$. If the singular value decomposition of \mathbf{G} provides \mathbf{V} and \mathbf{W} matrices such as $\mathbf{G} = \mathbf{W}\mathbf{D}\mathbf{V}^T$, then $\mathbf{G}\mathbf{G}^T = \mathbf{W}\mathbf{D}^2\mathbf{W}^T$ with $\mathbf{D} = \sqrt{\mathbf{D}_p}$ and $\mathbf{W} = \mathbf{T}$. Following the same reasoning, one has $\mathbf{V} = \mathbf{U}$.

The demonstration is achieved and the best correlated matrix \mathbf{A} with \mathbf{G} is the one satisfying $\mathbf{A} = \mathbf{T}\mathbf{E}\mathbf{U}^T$, with $\mathbf{G} = \mathbf{T}\mathbf{D}\mathbf{U}^T$.