Accuracy and speed assessment of 3D cross-correlation algorithms for two-frame and multi-frame PIV

Thomas Earl, Young Jin Jeon, Bertrand Lecordier, Laurent David

11TH INTERNATIONAL SYMPOSIUM ON PARTICLE IMAGE VELOCIMETRY - PIV15
Santa Barbara, California, September 14-16, 2015

Accuracy and speed assessment of 3D cross-correlation algorithms for two-frame and multi-frame PIV

Thomas Earl[1], Young Jin Jeon[1], Bertrand Lecordier[2] and Laurent David[1]

[1] Institut Pprime - ENSMA, UPR CNRS 3346, Université de Poitiers, Poitiers, France. thomas.earl@univ-poitiers.fr

[2] CORIA, UMR CNRS 6614, Université et INSA de Rouen, 76801 Saint Etienne de Rouvray Cedex, France

ABSTRACT

The need to extract higher order volumetric experimental data from PIV such as acceleration and pressure fields benefits from fast and accurate cross-correlation methods. This paper re-examines the application of two-dimensional (2D) cross-correlation methods to three-dimensional datasets by Bilsky et al. [3] and the binning techniques of Discetti et al. [7]. A robust version of the 2D methods, which reconstructs the three-dimensional (3D) signal from the 2D cross correlation maps is proposed. This method is then extended to the the fluid trajectory evaluation from ensemble-averaged cross-correlation (FTEE) method. Performance tests based on computational time and accuracy for both two-frame and multi-frame PIV were carried out on synthetically generated data, where the errors and correlation signal can be investigated. The cases presented herein include uniaxial uniform linear displacements and shear, and a final comparison of the FTEE method on synthetic homogeneous isotropic turbulence (HIT) data. For the cross-correlation computation, the proposed algorithm is in the order of 10 times faster than a standard 3D FFT. The FTEE method reduces the bias and random errors in the HIT data, and we find that the proposed algorithms yields almost identical results with significant speed-up.

1. Introduction

Volumetric PIV methods represent some of the state-of-the-art in experimental fluid mechanics research. Volumetric data are highly sought after, particularly for turbulent flows, which account for most environmental, because the study of turbulent structures can increase our fundamental understanding of their physical processes. In a two-frame single-exposure implementation, volumetric PIV requires three-dimensional (3D) particle reconstruction or localisation from the two sets of acquired imaged particles. Then, the calculation of a volumetric velocity field is conceptually identical to that of a planar or two-dimensional (2D) case: the two volumes of discretised particles are divided into interrogation regions (herein referred to as interrogation volumes, IV), and the probability of the pattern of particle displacement is computed by cross-correlation.

The drawback of 3D cross-correlation methods are that they are computationally expensive compared with their 2D counterparts. For example, consider a planar PIV 64×64 px interrogation window compared to an equivalent volumetric PIV $64 \times 64 \times 64$ vx IV. The 3D implementation requires approximately 100 times more calculations. In the case of tomographic PIV, the velocity field calculation typically requires more computational resources than the reconstruction of particle positions [2]. The cost of 3D cross-correlations is even greater when more than two time-steps are required (or desired) to produce one velocity field or higher order acceleration and pressure fields. Methods such as the sliding averaging correlation [15, 17], pyramid correlation [19], fluid trajectory correlation (FTC) [13] and fluid trajectory evaluation from ensemble-averaged cross-correlation (FTEE) [11] are examples of such schemes. The reported number of cross-correlations required N_{cc} for a sequence of N particle fields by these techniques are presented in Table 1.

Table 1: Number of cross-correlations N_{cc} for the reported number of sequential particle fields N. τ is the characteristic time-scale of the flow, i.e the largest time interval describing the smallest resolvable motions.

Technique	Reference	N	N_{cc}	Remarks
Sliding average	[15],[17]	3–4	$N-1$	$N < \tau$ or laminar, μPIV
Multi-frame pyramid	[19]	4–7	$\frac{1}{2} \cdot (N-1)^2$	$N < \tau$ or laminar, μPIV
FTC, FTEE	[13], [11]	5–9	$N-1$	Non-linear trajectory

Note that the values presented in Table 1 are for *one* iteration only. Typically more than two passes are required, encompassing window offset and deformation techniques to improve accuracy. To compound this further, to obtain appropriate convergence of high order statistical

moments in turbulent flows requires vast amounts of samples, often in the tens of thousands of realisations. Needless to say, fast and accurate PIV algorithms are highly desirable to reduce the burden of these computations.

The issue of speed has been addressed by several authors for two-frame single exposure PIV. Specifically focussing on 3D techniques, Discetti & Astarita [6] looked at redundant calculations (also considered in 2D by [16]), binning techniques and block algorithms. Bilsky et al [3] investigated the application of two dimensional (2D) techniques. The projection of the particles within an IV onto its sides for 2D cross-correlation, is of particular interest in this article. At the same time, Ziskin et al. (2011) [26] and later Brücker & Nonn (2012) [4] have considered 2D slices of the full volume of particles, employing 2D PIV techniques on orthogonal reconstructed planar images and then computing the 3D vectors from a combination of the 2D vector components.

The displacement of particles in two-frame, sliding-average, and multi-frame pyramid PIV is approximated by a linear vector. FTEE [11] is perhaps the most advanced of the 3D cross-correlation techniques. The technique relies on computing an ensemble-averaged cross-correlation map along a non-linear fluid trajectory, and was shown to be more robust with higher accuracy than FTC [13]. The motivation of this work is to accelerate the calculation of the FTEE method, while exploiting its reliance of the 3D ensemble average. We present a new and robust implementation of the 2D projection technique for volumetric PIV data that achieves this objective. We compare the performance of the algorithm with a range of other techniques to compare the accuracy and signal quality for linear displacements and simple shears. Finally, the computation of the FTEE method is performed on synthetically generated homogeneous isotropic turbulence data.

2. Algorithms

Before describing the accelerated algorithms used in this paper, we note that several techniques have been recently introduced for volumetric PIV, but not considered further. These include block algorithms [6] (for overlapping IVs, which is often the case), sparse calculations [2] (when particle density is low), and avoidance of redundant calculations [6]. Under the appropriate conditions these techniques can be feasibly applied to accelerate the calculation of the algorithms investigated herein. The following algorithms were coded in C++ using the open source library SLIP [22].

2.1 Algorithm: fft3d

Ignoring for the moment weighting functions and normalisation, the motor of the cross-correlation algorithm is a sliding dot product of one IV over the other:

$$C_{n+1} = \sum_{\mathbf{d}} \mathrm{IV}(\mathbf{x})_n \cdot \mathrm{IV}(\mathbf{x}+\mathbf{d})_{n+1}$$

where C is the cross-correlation map, $\mathbf{x}$ is the three-component vector of the IV voxel coordinates, $\mathbf{d}$ is the vector of incremental shifts in the three spatial directions, and n the time-step. The location of the maximum of this function corresponds to the most probable displacement of particles. The sliding dot product over the entire dimension of the IVs is computationally expensive (see Figure 2a) at the end of this section for the complexity of all presented algorithms) and is seldom used. Thus, the most commonly implemented 3D cross-correlation algorithm is achieved by taking advantage of the fast Fourier transform (FFT, or $\mathcal{F}()$). The cross-correlation map C by FFT is obtained in three steps: first, the discrete FFTs of two corresponding IVs are computed; second, the product of the complex conjugate of the FFT of IV_n and the FFT of IV_{n+1} is taken; third, the inverse FFT is taken of the product:

$$C_{n+1} = \mathcal{F}^{-1}\Pi\big(\mathcal{F}^*(\mathrm{IV}(\mathbf{x})_n)\mathcal{F}(\mathrm{IV}(\mathbf{x})_{n+1})\big)$$

where $\mathcal{F}^*$ indicates the complex conjugate. The 3D FFT cross-correlation algorithm, hereinafter fft3d, is used as the 'standard' algorithm for which to compare those presented in the following. The FFTs are computed with the FFTW library [9].

2.2 Algorithm: direct

The 'direct' cross-correlation is similar to a high accuracy final pass algorithm, as proposed by [6]. The direct algorithm is the dense sliding dot product computation with a limited displacement radius of 1 vx. Note that in this implementation, the resulting C of size $3\times3\times3$ vx is only filled along the orthogonal rows that pass through the central position of C. This reduces the number of dot products operations from $3^3 = 27$ to 7. The algorithm computation enables a three-point function fitting for the computation of the sub-pixel displacement. Therefore, this algorithm is only valid if the maximum component of the correction displacement vector $|\mathbf{s}_{\mathrm{corr}}| < 0.5$ vx, as when the correlation peak is not the central value, the peak fitting process will fail and introduce large errors.

2.3 Algorithm: binning

Binning techniques were first proposed by Discetti & Astarita (2012) [6]. Binning refers to the reduction of the IV by sampling clusters of voxels with a cubic kernel of size B, so that an IV window of 64^3 vx can be reduced to 32^3 with $B = 2 \equiv 2^3$ vx bin kernel. This substantially decreases the order of the number of operations required to compute the cross-correlation. In this implementation, the binning kernel is a mean operator. Other operators, such as a max operator, are not considered in this work. Additionally, we only consider the case where $B = 2$.

2.4 Algorithms: f2darb, f2dsnr

Bilsky et al, (2011) [3] proposed two techniques for accelerating the computation of a volumetric velocity field. The first technique reconstructs 3D cross-correlation maps from 2D cross-correlation maps, computed on the acquired PIV images themselves. The main idea of this technique was to avoid the need to reconstruct the volumes of particles (with a MART, for example), significantly saving computational cost. Their results showed that this technique was not robust, particularly with errors increasing in the depth of the particle volume (owing to their camera arrangement). This technique is not considered further for the following reasons. First, since the original contribution, the speed of tomographic reconstructions has increased, for example with multi-core BiMART [21], coarse to fine techniques [8], and time-resolved implementations [14]. Secondly, other techniques such as scanning TPIV [5] also offer very fast particle volume reconstruction. Thus the reconstruction of particles has become less of a bottleneck.

The second technique introduced by [3] involves the following procedure:

1. For the nth time step to be cross-correlated with time zero, the particles are projected onto the three orthogonal faces of the IV, resulting in a set of three 2D images $I_{\mathrm{p}_i,n}$ (Figure 1a), where the subscript p indicates an orthogonal projection and subscript $i \in 1,2,3$ the index of the three orthogonal planes of the IV.
2. Each corresponding projected image is cross-correlated, for example $I_{\mathrm{p}_i,0} \otimes I_{\mathrm{p}_i,n}$, resulting in three 2D cross correlation maps, $C_{\mathrm{p}_i,n}$ (Figure 1b).
3. Two 2D displacement vectors $\mathbf{s}_{\mathrm{p}_i,n}$ are calculated from $C_{\mathrm{p}_i,n}$.
4. The 3D displacement vector $\mathbf{s}_n$ is computed from a combination of the components of $\mathbf{s}_{\mathrm{p}_i,n}$ (Figure 1c).

In an effort to reproduce the results of Bilsky, two algorithms for the calculation of $\mathbf{s}_n$ from $\mathbf{s}_{\mathrm{p}_i,n}$ are tested. First, a direct component retrieval (Eq. 1), which is equivalent to Bilksy's original contribution and requires the minimal amount of computational operations. Two components of $\mathbf{s}_n$ are taken directly from one projected vector, and the remaining component from another projected vector. The choice of projections to use is arbitrary and fixed.

$$s_n^x = s_{\mathrm{p}_1,n}^x; \qquad s_n^y = s_{\mathrm{p}_1,n}^y; \qquad s_n^z = s_{\mathrm{p}_2,n}^z \tag{1}$$

Depending on the flow conditions, this may be optimised. This algorithm will be called f2darb, owing to its *arb*itrary computation of $\mathbf{s}_n$. The second approach is to take the velocity components from the 2D correlation maps with the maximum signal-to-noise ratio (SNR). The SNR is computed by comparing the highest with the second highest correlation peak.

$$s_n^x = \begin{cases} s_{\mathrm{p}_1,n}^x & \text{if } \Omega_1 > \Omega_2 \\ s_{\mathrm{p}_2,n}^x & \text{otherwise} \end{cases} \tag{2}$$

With similar expressions for s_n^y and s_n^z. The idea of this algorithm is to only utilise the best quality data in the retrieval of $\mathbf{s}_n$. This algorithm will be called f2dsnr. It should be noted that methods that incorporate calculating $\mathbf{s}_n$ from a mean of the 2D vectors $\mathbf{s}_{\mathrm{p}_i,n}$ are not desirable as erroneous components will make otherwise 'correct' components erroneous too.

2.5 Proposed Algorithm: f2dpr

The proposed algorithm, called fast 2D projection and reconstruction (f2dpr), combines the features of the two techniques proposed by [3]. The process of projecting the particle images and computing the 2D correlation maps is the same. However, in this algorithm the set of $C_{\mathrm{p}_i,n}$ are re-projected into a 3D volume of equivalent size with the minimum line-of-sight (MinLOS) algorithm commonly used to initialise tomographic PIV particle volumes. This operation results in a reconstructed 3D cross-correlation map, C_n (Figure 1d). The displacement vector $\mathbf{s}_n$ is then computed from the peak of C_n, with sub-pixel peak position interpolation, outlier detection and erroneous vector replacement following standard 3D techniques.

The idea of the reconstruction is to utilise all of the available information from $C_{\mathrm{p}_i,n}$ to increase the signal quality of the stand-alone 2D methods, and to use all of the possible information of the three 2D cross-correlation maps in a robust way. As seen, f2darb uses only two of the cross-correlation maps to compute $\mathbf{s}_n$ and f2dsnr requires an additional computation of the SNR. As mentioned in the introduction, the reconstructed cross-correlation signal can be employed in techniques such as FTEE, which enhances the robustness again, compared to two-frame PIV techniques. This feature is the key driver of this technique.

3. Comparison

3.1 Description of synthetic volumes

We use synthetically generated particle intensity volumes are used to investigate and quantify errors, speed and signals of the algorithms. Particles were randomly distributed and then discretised into voxel intensities of Gaussian energy distribution of mean diameter of 3 vx. The final discretised volumes were $512 \times 512 \times 256$ vx in dimension. Unless stated otherwise, the equivalent particles-per-pixel (ppp, for an equivalent 512×512 px image) was 0.05, or an image source density of $N_s = 0.7$, typical values of most reported tomographic PIV

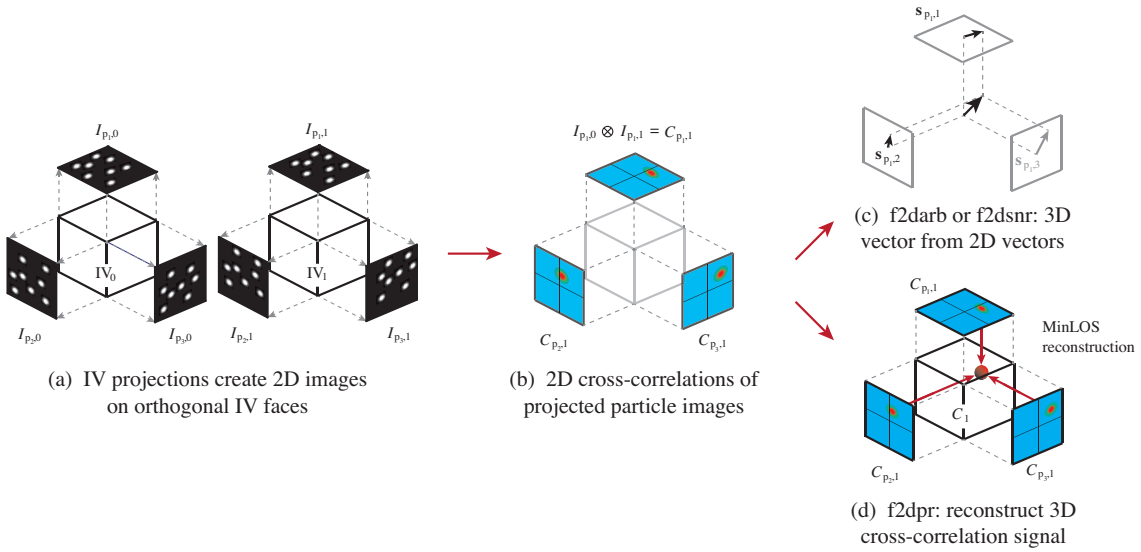


Figure 1: Calculation of 3D displacement vector from 2D projections. (a) Particles in each interrogation volume IV are projected onto two (or three) orthogonal sides, (b) calculation of two (or three) 2D cross-correlation maps [3], (c) combination of 2D displacement vectors to achieve 3D displacement vector, (d) reconstruction of correlation signal by MinLOS to find 3D displacement vector.

experiments. All values quoted in this section are averaged over the total number of IVs computed in the particle volumes, which varied with IV size. The aim of this section is to address the speed-up of the proposed algorithm and the properties of the resulting signal.

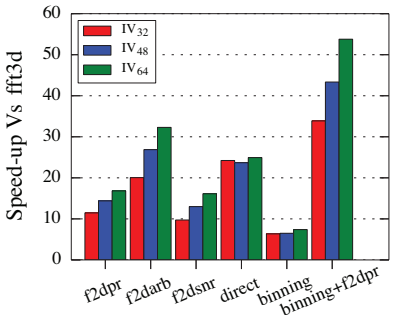
3.2 Speed-up and complexity of algorithms

The f2dpr can naturally be extended to voxel binning [6] (here denoted binning+f2dpr) where an even greater increase in speed is possible, is also investigated. The complexity of all the algorithms is presented in Figure 2a. The complexity of the algorithms tends to reduce moving down the table, however it is not immediately obvious which are the most efficient of the accelerated algorithms. The speed-up compared to fft3d is shown in Figure 2b for three IV sizes. The speed-up includes all overheads of the cross-correlation, such as the creation or filling of binning containers or temporary arrays for the 2D projection algorithms, until the computation of $\mathbf{s}$. A significant speed-up by all algorithms is observed, the highest obtained when combing the binning and f2pdr algorithm, which results in a speed-up of up to 53$\times$ for a 64^3 vx IV. These speed-ups are inline with those presented by [4].

Considering the 2D projection algorithms, it is interesting to note that the f2darb algorithm is slightly slower than f2dpr. This indicates that the projection and re-projection stages are fast compared to the three SNR calculations. Considering f2darb, we deduce that the 2D cross-correlations are the most significant computations, as f2darb is approximately one third faster than the other two 2D projection methods as it computes only two of the three correlations. The speed-up for the binning algorithm is somewhat slower than the 2D methods, owing to the expensive 3D cross-correlation. The speed-up for the binning is slightly slower than the quoted value of 10 times in the original contribution [6]. This could be due to implementation differences and efficiencies in the binning process.

Algorithm	Cross-correlation	Overheads
sliding dot product	$o(W^6)$	—
fft3d	$o(W^3 \cdot \log_2(W^3))$	—
f2dpr	$o(W^2 \cdot \log_2(W^2))$	$o(W^3)$
f2darb	$o(W^2 \cdot \log_2(W^2))$	$o(W^2)$
f2dsnr	$o(W^2 \cdot \log_2(W^2))$	$o(W^2)$
direct	$o(D^3W^3)$	—
binning	$o(W^3/B^3 \cdot \log_2(W^3/B^3))$	$o(W^3/B^3)$
binning+f2dpr	$o(W^2/B^2 \cdot \log_2(W^2/B^2))$	$o((W^3/B^3)^2)$

(a)



(b)

Figure 2: (a) Table of computational complexity (b) Speed-up of algorithms compared to standard implementation of fft3d. The speed for three cubic IV sizes are shown, where the subscript in the legend indicates its linear dimension W.

Table 2: Conversion of ppp metric for Figure 4

ppp	ppv	$N_{\mathrm{p,IV}}$	$N_S = \mathrm{ppp} \cdot \pi \mathrm{p}_d^2/4$	$N_{S,\mathrm{IV}} = \mathrm{ppv} \cdot 4\pi \mathrm{p}_d^3/3$
0.02	0.0002	6.5	0.13	0.02
0.05	0.0005	17	0.33	0.06
0.1	0.001	36	0.74	0.12
0.2	0.002	70	1.00	0.24

3.3 Errors on uniform displacements

Figure 3 shows the bias errors β and random errors ε (Eq. 3) for the algorithms investigated for a given imposed displacement along the x-direction, x_{imp}, between 0 and 3 vx at 0.1 vx increments. The errors are computed on the l^2-norm of the displacement vector for the cross-correlation first pass only.

$$\beta = \frac{1}{N}\sum\left(|u_{i,\mathrm{piv}} - u_{i,\mathrm{dns}}|\right) \tag{3a}$$

$$\varepsilon = \sqrt{\frac{1}{N}\sum\left(|u_{i,\mathrm{piv}}| - |u_{i,\mathrm{dns}}|\right)^2} \tag{3b}$$

It is observed that the peak β for the binning and binning+f2dp algorithms are in the order of 4 times greater than the other algorithms. A similar observation can be made for ε, although here the difference is closer to an order of magnitude. The form of the binning algorithm curves are also different: their period is twice (or a multiple of B times) that of the other correlation techniques, arising from the multiplication of the displacement vector in the binned IV $\mathbf{s}_{\mathrm{bin}}$ to return the correct displacement, $\mathbf{s} = B \cdot \mathbf{s}_{\mathrm{bin}}$. Notable also is that all 2D projection methods are almost identical to the fft3d algorithm and direct cross-correlation method (with range of displacements limited between 0 and 0.5 vx). Therefore, for synthetic data, the application of f2dpr is feasible, returning similar results to the standard fft3d algorithm. The errors of the binning techniques, however, are too large. The errors presented here are inline with recent studies, such as [13, 11]. The evolution of the errors is slightly different however, with sharp peaks at the half pixel intervals. This may be due to the parabolic sub-pixel detection method used in this paper compared to a Gaussian fit or a peak locking effect.

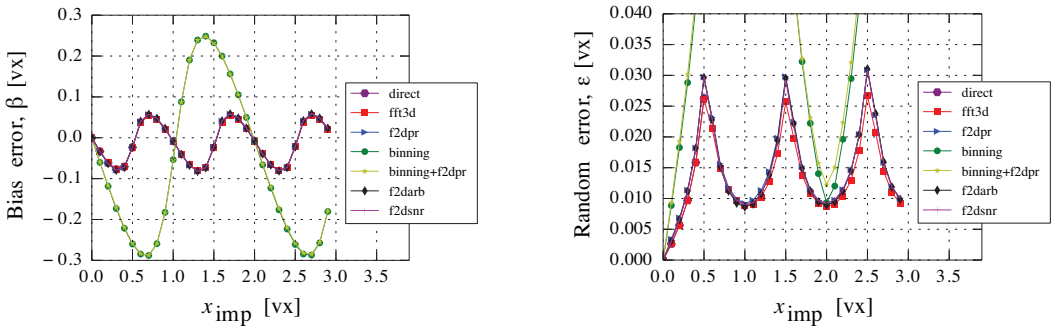
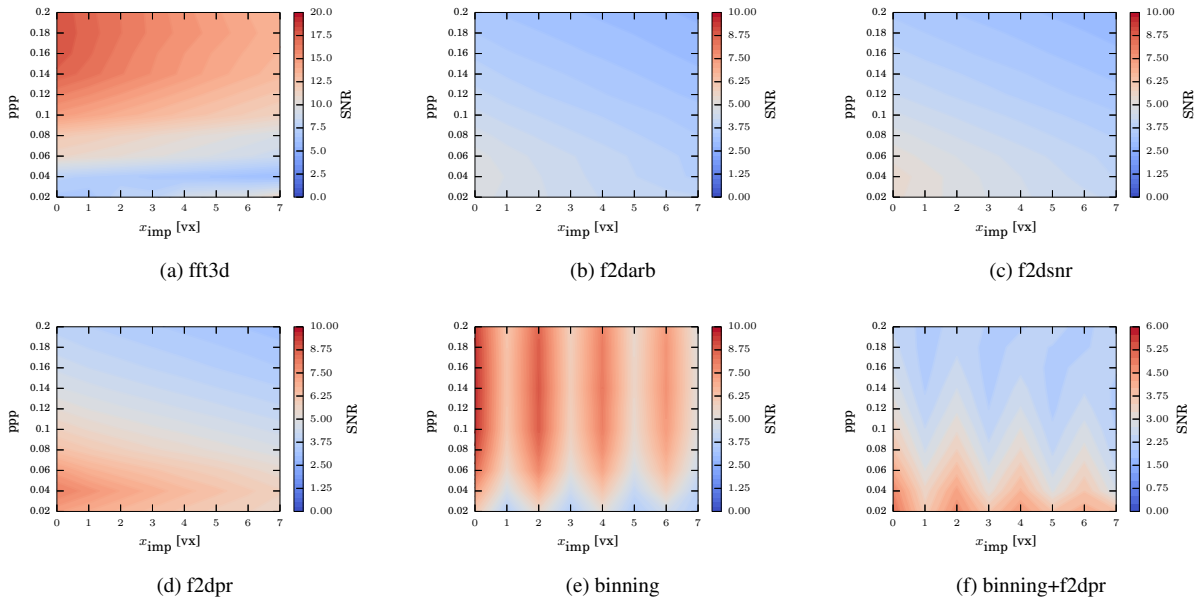


Figure 3: Bias β and random ε errors for all of the algorithms for linear displacements in the x direction (x_{imp}). The peak values of ε for both binning implementations peak at 0.25 px at integer values of x_{imp}.

3.4 Signal-to-noise ratio

Figure 4 shows the SNR of all of the algorithms (except for direct) with respect to ppp and imposed displacement (x_{imp}), in order to find the limitations of the algorithms. As noted by other authors (e.g. [21]), the ppp is not an ideal measurement of particle density, so Table 2 shows the salient values of ppp expressed as particles per volume (ppv), image source density (N_S), volume source density ($N_{S,\mathrm{IV}}$), and average number of particles in the IV ($N_{\mathrm{p,IV}}$). It is not possible to retrieve a SNR from the direct algorithm, as it is computed only a central $3 \times 3 \times 3$ kernel centred on previous iterations. The SNR was calculated in IVs of 32^3 vx. Note that the colour bar values for fft3d (Figure 4a) is twice and the binning+f2dpr is half that of the other algorithms. We also note that the SNRs presented here are higher than what would be achieved for experimental measurements, owing to the lack of experimental noise and reconstruction artefacts.

Here we clearly observe the trade-off of the accelerated measurements: reduction in signal quality. All acceleration algorithms have a significant drop in signal strength compared to fft3d. This is largely a result of a reduction in spatial resolution. For the 2D projection algorithms, particles are projected onto an IV face, and the spatial depth dimension is lost. The higher SNR for f2dpr indicates that



(a) fft3d (b) f2darb (c) f2dsnr

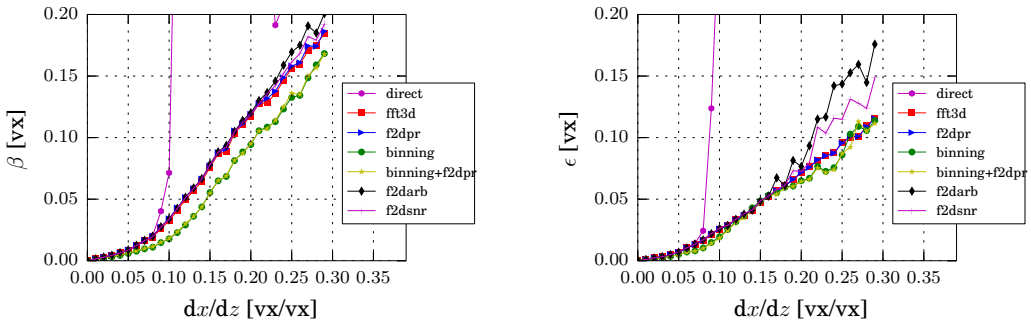
(d) f2dpr (e) binning (f) binning+f2dpr

Figure 4: Signal to noise ratio (SNR) contour plots of the algorithms with respect to imposed linear displacement and particle density. The corresponding algorithms are noted in the sub-captions. Note that the maximum of the colour bar is double for fft3d (a) and half for binning+f2dpr (b).

reconstruction of the peak partially regains this loss. The effect of the binning kernel is clearly visible in Figure 4c and Figure 4d. At odd number pixel displacements, there is an approximate 40% drop of SNR over the range of ppp. This corresponds well with β and ε errors, where the lack of signal strength corresponds to higher errors. We note here that, although mean SNR of the f2dpr and binning algorithms are similar, this does not have a direct correlation with the errors (see previous section). The fastest algorithm, the binning+f2dpr has a very poor signal quality compared to the others shown.

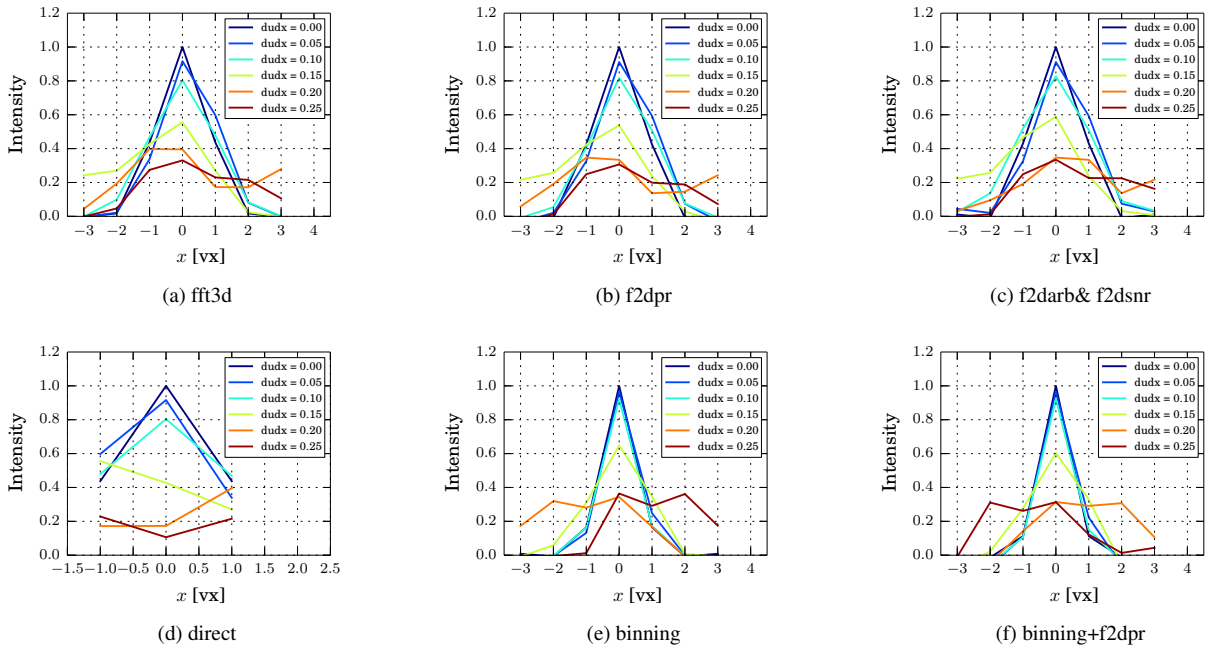
3.5 Linear shear flow

To further the performance analysis of the algorithms, a linear shear dx/dz is applied to the particle locations in a synthetic volume. Here, the volume chosen is one IV thick in z (namely 32 vx), with 1024 non-overlapping IVs . Therefore the statistics shown Figure 5 are averaged quantities with pure shear only (i.e. with zero mean displacement). As discussed in by [23, 20] through analytical analysis, a pure shear tends to broaden and reduce the correlation peak (reduce the SNR), resulting in increased random errors. Additionally, the broadened peak is no longer symmetrical, introducing bias errors. This behaviour is observed for all algorithms tested (Figure 5). Notably, the proposed f2dpr algorithm performs equivalently to the fft3d. The f2darb and f2dsnr have greater β and ε from gradients of 0.25 px/px, approaching double the errors of fft3d and f2dpr for the maximum gradient tested. For the linear shear case, the binning algorithms perform better than f2darb and f2dsnr algorithms, contrary to the linear displacement case presented earlier in this section. The direct algorithm however suffers from significantly higher bias errors from dx/d$z \approx 0.10$. This effect was investigated further, and it was found that as the correlation peak broadened, the peak of the signal was increasingly not in the central position of the $3 \times 3 \times 3$ vx search window. When this is the case, the sub-pixel peak interpolation fails significantly, sometimes computing the peak several voxels away from the centre position. On the other hand, the other algorithms are more stable as the cross-correlation map is computed over the entire IV.



(a) (b)

Figure 5: Results of simple shear within the IV. (a) the bias error β and (b) the random error ε of the l^2-norm of the velocity fields.



(a) fft3d (b) f2dpr (c) f2darb& f2dsnr

(d) direct (e) binning (f) binning+f2dpr

Figure 6: 1D cross section of C at the its centre in the direction of simple shear. The corresponding algorithms are noted in the sub-captions.

One-dimensional plots of the normalised cross-correlation peaks for single IVs are shown in Figure 6. Note that for each imposed gradient, a unique particle field was used for all algorithms. Therefore, the correlation signals do not correspond to the same particle distribution. The cross-correlation peaks of the fft3d and f2dpr algorithms are very similar, helping to explain their near identical β and ε errors. For the f2darb & f2dsnr algorithms (Figure 6c), it is observed that the peak is broader than fft3d and f2dpr for gradients of 0.15 vx/vx, again corresponding to the errors presented in Figure 5. The binning algorithms reduce the diameter of the correlation peak, as the particle diameter in the binned IV have been divided by $B = 2$, the binning kernel. The binning also acts to reduce the effective gradient imposed on the IV by B. So the broadening of the peak for these algorithms occurs more gradually compared to the fft3d algorithm up until a gradient of 0.15 vx/vx, where the peak is still quite symmetrical. This effect explains the reduced β and ε observed in Figure 5. At the highest gradients, double peaks are observed. As discussed previously, these caused an instability in the direct algorithm sub-pixel interpolation.

3.6 Features of f2dpr reconstructed signal

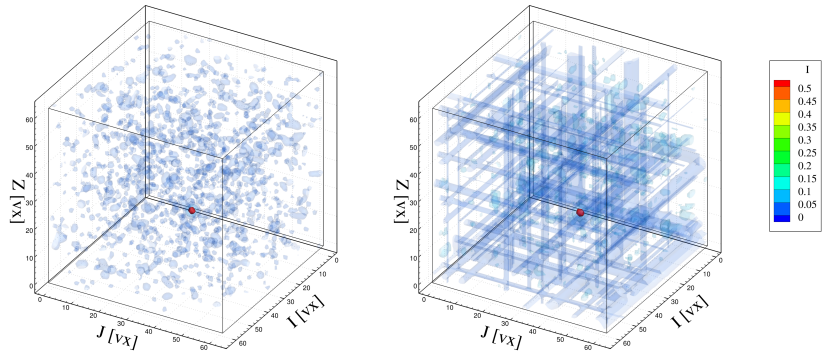
In order to validate the application of f2dpr to FTEE, it is important to consider the effects of reconstructing the 3D cross-correlation signal. Figure 7 shows visualisations of the normalised 3D cross-correlation maps C_n for the fft3d and f2dpr algorithms for a imposed displacement. Iso surface levels are set to 0.02, 0.1, and 0.5 to show the distribution of the noise in the volume (blue and turquoise) as well as the location of the peak (red). As one may expect, the MinLOS reconstruction generates tube like artefacts: as I_{p_i} are projected into the volume, the low intensity signals traverse the full length of the IV. In this example, the peak signals are clearly distinguishable for both algorithms. As mentioned, the similarity of the peaks is an important feature of the f2dpr, as this allows the utilisation of the multi-frame techniques.

4. FTEE test case

This previous section showed that for a range of tests, the f2dpr algorithm is superior to the other accelerated algorithms with equivalent errors to a standard fft3d implementation. It indicated that the f2dpr algorithm could be suitable in both two-frame PIV and also implementations that rely on the ensemble average to compute trajectories, such as with FTEE, not just for predictor fields, but for computing the final velocity field. The current section compares the performance of f2dpr against fft3d for a synthetically generated particle field displaced by a turbulent velocity field.

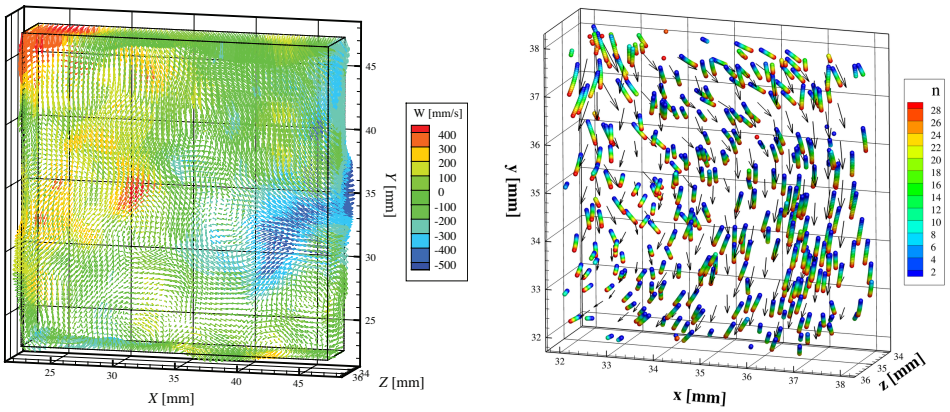
4.1 Synthetic turbulence data

In order to compare the performance of f2dpr using FTEE, a set of synthetic 3D particle fields are displaced by a stationary/singular direct numerical simulation (DNS) [12] field simulating homogeneous isotropic turbulence (HIT) (Figure 8). The DNS field displaces the particles in a pseudo-time-resolved mode: the particles are subjected to the same velocity field in a stepwise manner so that they obtain a Langrangian motion through the volume. The stepwise displacements adhere to the Kolmogorov timescale limit of the velocity field ($\Delta t < \tau_\eta$), so that the linear trajectories that are imposed between each time-step do not truncate the kinetic energy of the system.



(a) fft3d (b) f2dpr

Figure 7: Normalised 3D cross-correlation maps of (a) the standard fft3d algorithm, and (b) the f2dpr algorithm. The correlation peak is the red point. Iso-surfaces are at I = 0.02 and I = 0.1. The MinLos reconstruction in f2dpr is evident through propagation (tubes) of low intensity.



(a) (b)

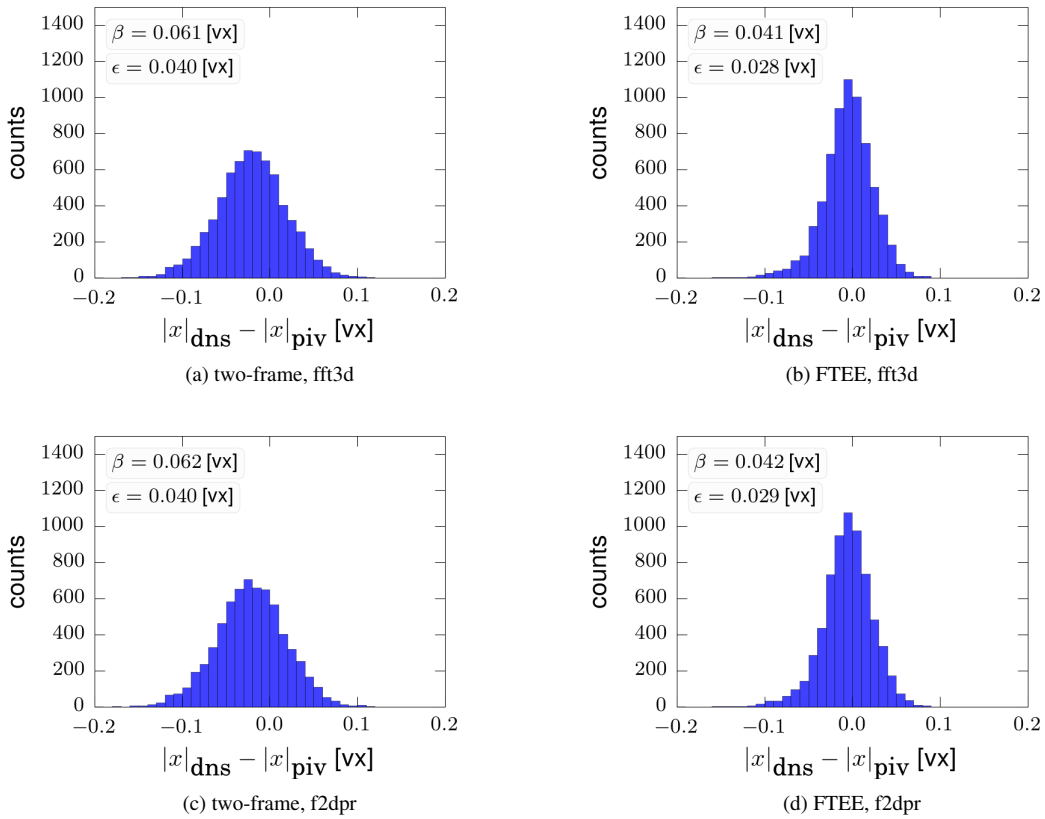
Figure 8: Visualisation of HIT velocity field used to displace particles in both two-frame and pseudo-time-resolved modes. (a) full velocity field, (b) zoom of particles coloured by time to show trajectories.

The particle density is based on generating images of ppp $= 0.06$ ($N_S \approx 0.5$), with discretised particle diameters of 3 vx with Gaussian intensity. The final discretised size of the volume was $512 \times 512 \times 128$ vx, with IVs of 32^3 vx with 3D tophat weighting. The size of the voxels was chosen so that the centre-to-centre spacing $\delta x = 3\eta$, where η is the Kolmogorov lengthscale, with 50% overlap. To enable comparison with the PIV algorithms, the DNS field was filtered with a mean operator uniform weighting window of equivalent size to the IV so that the differences with the algorithms are independent of the averaging effect of the IVs [25, 1].

Seven sequential particle volumes I_n separated by Δt, where $n = -3,-2,\ldots,2$, were generated. The mean of the l^2-norm of the particle displacement $\overline{|\mathbf{s}|} \approx 5$ vx, and the max $|\mathbf{s}|_{\max} \approx 10$ vx from $n = -3$ to $n = 3$. The FTEE algorithm used all seven images to compute a trajectory from the ensemble averaged cross-correlation maps. The polynomial describing the trajectory $\Gamma(\mathbf{x},n)$ was 2nd order, where $\mathbf{x}$ is the vector of coordinates of the IV centres. Taking the first derivative of the trajectory at $n = 0$ yields the displacement vector, $\mathbf{s} = \mathrm{d}/\mathrm{d}n(\Gamma(\mathbf{x},0))$. The two-frame implementation is computed over the trajectory extremes, $I_{-3} \otimes I_3$. Both algorithms used 64^3 vx IVs for initialising a predictor velocity field, before switching to five iterations of 32^3 vx IVs with deformation [10, 18] allowing for full convergence. There was no post-treatment of the velocity field, only intermittent smoothing and median filtering [24].

4.2 Results

Figure 9 shows histograms of the l^2 normal differences between the DNS field and the two-frame and FTEE cross-correlation results. Imbedded in each figure are the bias and random errors from Equation 3. The figure highlights the gain in accuracy between the two-frame and FTEE approach. There is a 30% reduction in β and 25% reduction in ε, resulting in narrower histograms centred on the exact solution. Most importantly for this article, Figure 9 shows that the difference between fft3d and f2dpr are negligible, returning errors no worse than 10^{-3} vx. The greatest difference between the algorithms is with computation time. As discussed in section 3.1, f2dpr is about 10 times faster



(a) two-frame, fft3d

(b) FTEE, fft3d

(c) two-frame, f2dpr

(d) FTEE, f2dpr

Figure 9: Comparison of two-frame and FTEE implementation for synthetic HIT data. *left column*: two-frame PIV *right column* FTEE

than fft3d for IVs of 32^3 vx. The ultimate speed-up for the complete calculation of a velocity field will depend on the optimisation of the interpolation scheme used for the deformation algorithms, as these calculations are of a similar order to the cross-correlations. We recorded a speed-up of ≈ 2 for the FTEE using f2dpr compared to fft3d without an optimised spline scheme. Although still a significant improvement in computational time, the total speed-up could be improved further with some modest improvements to the spline deformation scheme.

5. Conclusion and outlook

An extensive analysis of the speed and accuracy of cross-correlation methods for volumetric PIV data has been presented. For the calculation of large experimental turbulence data sets from volumetric PIV data, an effort to increase the speed of cross-correlation is highly desirable. A robust implementation of 2D cross-correlation methods, called f2dpr, was introduced. The algorithm reconstructs a 3D cross-correlation signal from three orthogonal 2D cross-correlation maps for each IV. This algorithm was designed with the FTEE method in mind, as it uses a sequence of particle fields (increasing computational time) and it computes particle path trajectories from the 3D ensemble averaged cross-correlation maps.

The performance of the f2dpr algorithm based on computational time and accuracy was presented for a variety of synthetic cases and against a variety of other algorithms. For linear displacements, the bias and random errors of the binning algorithms were significantly greater than the other algorithms tested, while the f2darb and f2snr had significantly reduced SNRs. For a linear shear, the f2darb and f2snr algorithms had inferior performance at higher shear levels. It was seen that the signal peak of f2dpr was similar to the fft3d. The f2dpr algorithm was found to have an appropriate balance of speed and accuracy for full velocity field calculation.

The f2dpr algorithm was implemented into the FTEE technique using synthetic HIT synthetic data. For the presented case, f2dpr performed almost identically to the standard fft3d algorithm in terms of accuracy with a significant reduction in total computational time. It was also shown that there was a significant gain in accuracy of the FTEE versus a two-frame cross-correlation baseline. Therefore, the f2dpr algorithm is a promising technique for producing high accuracy results, as shown by the synthetic cases.

Future investigations will consider experimental tests cases and will vary the relocation and trajectory length of the synthetic HIT case, in order to find any stricter limitations compared to the fft3d. In addition, although it was demonstrated that f2darb and f2snr underperformed compared to f2dpr, it is intended to implement them into the FTEE method to test their stability and accuracy. The application of the direct algorithm to the 2d projected images could also add a significant increase to speed for the final iterations of the cross-correlation.

References

[1] C H Atkinson, N A. Buchmann, O Amili, and J Soria. On the appropriate filtering of PIV measurements of turbulent shear flows. *Experiments in Fluids*, 55(1):1654, 2014.

[2] C H Atkinson and J Soria. An efficient simultaneous reconstruction technique for tomographic particle image velocimetry. *Experiments in Fluids*, 47(4-5):553–568, August 2009.

[3] A V Bilsky, V M Dulin, V A Lozhkin, D M Markovich, and M P Tokarev. Two-dimensional Correlation Algorithms for Tomographic PIV. In *9TH International Symposium on Particle Image Velocimetry - PIV11 Kobe, Japan, July 21-23, 2011*, pages 1–4, 2011.

[4] C Brücker and T Nonn. Rapid 3D velocity reconstruction from volumetric particle fields via re-slicing and recombination. In *16th Int Symp on Applications of Laser Techniques to Fluid Mechanics. Lisbon, Portugal, 9-12 July*, pages 1–10, 2012.

[5] L. David, T. Jardin, P. Braud, and A. Farcy. Time-resolved scanning tomography PIV measurements around a flapping wing. *Experiments in Fluids*, 52(4):857–864, June 2011.

[6] S Discetti and T Astarita. A fast multi-resolution approach to tomographic PIV. *Experiments in Fluids*, 52(3):765–777, May 2011.

[7] S Discetti and T Astarita. Fast 3D PIV with direct sparse cross-correlations. *Experiments in Fluids*, 53(5):1437–1451, September 2012.

[8] S Discetti, Isaac B Ziskin, Tommaso Astarita, Ronald J Adrian, and Kathy P Prestridge. PIV measurements of anisotropy and inhomogeneity in decaying fractal generated turbulence. *Fluid Dynamics Research*, 45(6):061401, December 2013.

[9] M Frigo and S G. Johnson. Fastest Fourier Transform in the West (FFTW), www.fftw.org, 2014.

[10] H.T. Huang, H.E. Fiedler, and J.J. Wang. Limitation and improvement of PIV. *Experiments in Fluids*, 15-15(4-5):263–273, 1993.

[11] Y J Jeon, L Chatellier, and L David. Fluid trajectory evaluation based on an ensemble-averaged cross-correlation in time-resolved PIV. *Experiments in Fluids*, 55(7):1766, June 2014.

[12] B Lecordier, D Demare, L M J Vervisch, J Reveillon, and M Trinite. Estimation of the accuracy of PIV treatments for turbulent flow studies by direct numerical simulation of multi-phase flow. *Measurement Science and Technology*, 12:1382–1391, 2001.

[13] K P Lynch and F Scarano. A high-order time-accurate interrogation method for time-resolved PIV. *Measurement Science and Technology*, 24(3):035305, March 2013.

[14] K P Lynch and F Scarano. An efficient and accurate approach to MTE-MART for time-resolved tomographic PIV. *Experiments in Fluids*, 56:1–16, 2015.

[15] C D Meinhart, S T Wereley, and J G Santiago. A PIV Algorithm for Estimating Time-averaged Velocity Fields. *Journal of Fluids Engineering*, 122:285–289, 2000.

[16] G I Roth and J Katz. Five techniques for increasing the speed and accuracy of PIV interrogation. *Measurement Science and Technology*, 12:238–245, 2001.

[17] F Scarano, K Bryon, and D Violato. Time-resolved analysis of circular and chevron jets transition by tomo-PIV. In *15th Int Symp on Applications of Laser Techniques to FLuid Mechanics*, number 2004, pages 1–8, 2010.

[18] F Scarano and M L Riethmuller. Advances in iterative multigrid PIV image processing. *Experiments in Fluids*, pages 51–60, 2000.

[19] A Sciacchitano, F Scarano, and B Wieneke. Multi-frame pyramid correlation for time-resolved PIV. *Experiments in Fluids*, 53(4):1087–1105, July 2012.

[20] J Soria. Effect of Velocity Gradients on 3D Cross-correlation PIV Analysis. In *14th Int Symp on Applications of Laser Techniques to Fluid Mechanics. Lisbon, Portugal, 7-10 July*, pages 07–10, 2008.

[21] L Thomas, B Tremblais, and L David. Optimization of the volume reconstruction for classical tomo-PIV algorithms (MART , BIMART and SMART): synthetic and experimental studies . *Measurement Science and Technology*, 25:15pp, 2014.

[22] B Tremblais, L David, D Arrivault, J Dombre, L Chatellier, and L Thomas. SLIP: Simple Library for Image Processing (version 1.0), http://www.sic.sp2mi.univ-poitiers.fr/slip/, 2010.

[23] J. Westerweel. On velocity gradients in PIV interrogation. *Experiments in Fluids*, 44(5):831–842, December 2008.

[24] J Westerweel and F Scarano. Universal outlier detection for PIV data. *Experiments in Fluids*, 39(6):1096–1100, August 2005.

[25] C E Willert and M Gharib. Digital particle image velocimetry. *Experiments in Fluids*, 10:181–193, 1991.

[26] I B Ziskin, R J Adrian, and K Prestridge. Volume Segmentation Tomographic Particle Image Velocimetry. In *9TH International Symposium on Particle Image Velocimetry - PIV11 Kobe, Japan, July 21-23, 2011*, volume 17, pages 2–5, 2011.