Enhancing the velocity dynamic range and accuracy of time-resolved PIV through fluid trajectory correlation

Kyle Lynch¹,*
Fulvio Scarano¹

1: Department of Aerospace Engineering, Delft University of Technology, Delft, Netherlands

* correspondent author: k.p.lynch@tudelft.nl

Abstract A novel method is introduced for increasing the accuracy and extending the dynamic range of time-resolved PIV. The approach extends the well-known concept of multiframe particle tracking velocimetry (PTV) to cross-correlation analysis employed by PIV. The working principle is based on the determination of fluid element trajectories by tracking their position across an image sequence. The fluid trajectory correlation (FTC) algorithm deals with the effect of trajectory curvature and non-uniform velocity during the considered time interval by allowing the motion within the trajectory to be nonlinear. In addition, the local image deformation accounts for the spatial velocity gradient and its change along the trajectory.

The principle for reduction of the measurement error is threefold: by enlarging the temporal measurement interval, the relative error becomes smaller; secondly, the random error is reduced with the use of a least-squares polynomial fitting approach to the individual trajectory; and finally, the use of nonlinear fitting functions allows for a reduction in truncation errors. The evaluation of velocity proceeds then directly from the analytical derivatives of the least-squares functions. The principal features of this algorithm are compared with a single-pair iterative image deformation method through the use of synthetic image sequences depicting steady flows (solid body rotation and uniform motion), and with an application to an experimental data set of a submerged circular jet.

1. Introduction

Time-resolved PIV (TR-PIV) is a technique for the measurement of both spatial and temporal fluctuations in velocity, allowing for analyses in both the spatial and temporal domain. Proper application of the technique requires a sequence of particle images to be acquired at a frequency greater than a relevant timescale of interest in the flow. For many investigations, this requires the use of high-speed cameras equipped with CMOS sensors for imaging and high-repetition-rate lasers for illumination. In contrast to CCD sensors, CMOS sensors typically have larger pixels, exhibit higher noise, and have lower sensitivity (Hain et al., 2007; Hain and Kähler, 2007, among others). Additionally, high-repetition-rate, diode-pumped Nd:YLF lasers deliver approximately an order of magnitude lower pulse energy compared to low-repetition-rate, flashlamp-pumped Nd:YAG lasers. The combination of lower image quality and weaker illumination result in a considerable reduction of the image quality and lead to an increased uncertainty of the velocity measurements.

The potential of using image sequences in PIV image analysis has already been recognized in a number of studies. One of the earliest examples is given by Fincham and Delerche (2000), who used an initial measurement at a short Δt as a predictor for correlations performed at a larger time separation to increase the dynamic range of the measurement. This concept was refined by Hain and Kähler (2007) who proposed a method based on an optimal Δt approach in a manner similar to Pereira (2004). In brief, this method selects a specific pair of images within a multiframe sequence to increase the effective Δt in regions of low displacement, thus reducing the relative error in these regions. The choice of the optimal time separation is made locally on the basis of criteria that minimize the error estimate. The work of Persoons and O’Donovan (2011) also employs a similar approach, but with a different criterion on the optimal Δt.

A separate approach, termed sliding averaged correlation (SAC), approaches the topic by reducing random error through combining multiple correlations over a set of adjacent image pairs. Combining correlation planes has been demonstrated to increase the robustness of average velocity
estimates (Meinhart et al., 2000) and is currently widely practiced. In time-resolved unsteady flows where the acquisition rate is sufficiently high, correlation map averaging over a sliding kernel can be applied. A preliminary analysis of the method on experimental data was performed by Scarano et al. (2010), where a reduction of random noise was observed. A further application of the technique on pulsatile flows was performed by Poelma and Westerweel (2010). The SAC technique increases accuracy from the opposite perspective of multiframe PIV by reducing the random error $\epsilon_r$ through the summation of multiple correlations. However, the amount of reduction available is limited in flows with large curvature or acceleration due to the lower number of images which can be used.

A more recent approach is the pyramid correlation (Sciachitano et al., 2012), which combines the concepts of $\Delta t$ extension from multiframe PIV and the localized ensemble correlation from SAC. Briefly, the pyramid correlation technique performs a series of correlations over all possible combinations within a small kernel about a central image. For example, a central image $I_3$ will result in correlations $I_1 \otimes I_2$, $I_2 \otimes I_3$, $I_3 \otimes I_4$, and also the higher-level correlations $I_1 \otimes I_3$, $I_2 \otimes I_4$ and $I_3 \otimes I_4$. The effective $\Delta t$ of the correlation is then equal to the maximum time separation within the kernel. Additionally, the correlations can be merged together to form a single correlation plane by performing a linear transformation, or homothety, in order to properly scale the correlation planes to the effective $\Delta t$. The resulting error is inversely proportional to the number of correlations performed, or $\epsilon_r \propto \sigma_v/\sqrt{(N^2 + N)/2}$. Unfortunately, like SAC, the error reduction is limited in flows with curvature or acceleration.

The remainder of this paper proposes an alternative, novel method to achieve enhancements in accuracy of TR-PIV image analysis by adapting correlation-based algorithms to explicitly account for the effects of curvature and accelerations through an image sequence. First, a detailed description of the algorithm is given. This is followed by tests on synthetic data to independently assess the performance of the algorithm for reducing bias errors and random errors. The algorithm is then used to process experimental images of a circular jet to ensure the predicted improvements also apply for real data.

2. Fluid Trajectory Correlation Concept

The fluid trajectory correlation (FTC) technique is intended as a corrector for time-resolved velocity fields obtained through the application of traditional PIV algorithms. A description of the algorithm parallels that of image correlation velocimetry, a single-pair interrogation algorithm proposed by Tokamaru and Dimotakis (1995). The mapping between images of a sequence can be described by the continuous Lagrangian displacement field $\Gamma(x, t)$ throughout the sequence. In this description, the expression $\Gamma(x, t)$ represents the transport of an image coordinate, i.e., pixel, $x$ throughout the image sequence. Since $\Gamma(x, t)$ represents a position in time, the velocity can be determined through differentiation,

$$v[\Gamma(x, t), t] = \frac{d}{dt} \Gamma(x, t)$$  \hspace{1cm} (1)

Because $\Gamma(x, t)$ is unknown but the images $I$ are known, the focus of algorithms is to find the optimal form of $\Gamma(x, t)$ which allows these parameters to be estimated. To arrive at this, we first present a Lagrangian description of how the flow, and thus image, will vary in time. Equation 2 is the transport equation of a scalar quantity, in this case the image intensity $I$, without the influence of diffusion as can be assumed for particle images.

$$\frac{dI(\Gamma(x,t))}{dt} + u \cdot \nabla I(\Gamma(x,t)) = 0$$  \hspace{1cm} (2)

This indicates that the transport of the image intensity is completely described by the continuous
Lagrangian displacement field. The challenge is to determine \( \mathbf{\Gamma}(\mathbf{x}, t) \) in a manner which best approximates equation 2. By integrating this equation in time, it is possible to reframe the problem as an optimization. For example, with a single-pair analysis,

\[
\min_{\mathbf{\Gamma}(\mathbf{x}, t)} \int_{\Omega} \left( E(\mathbf{\Gamma}(\mathbf{x}, t_1)) - E(\mathbf{\Gamma}(\mathbf{x}, t_0)) \right) d^2 \mathbf{x}
\]

(3)

Due to the complex dynamics that can potentially be described by \( \mathbf{\Gamma}(\mathbf{x}, t) \), this optimization is particularly difficult. For single-pair evaluation, the complexity is limited and optimizers can be directly applied, e.g., Tokamaru and Dimotakis (1995) and Meyer (2002). Another simplification is used in traditional PIV algorithms, where the correlation operator is used as an approximate form of the optimization by which the peak of the correlation plane corresponds to the optimal match between two image patterns within the domain of linear shifts. The correlation procedure effectively restricts the form of \( \mathbf{\Gamma}(\mathbf{x}, t) \) to a first-order approximation of the Lagrangian displacement field in time.

If an image sequence is considered instead of a single pair, the complexity of \( \mathbf{\Gamma}(\mathbf{x}, t) \) is too large to apply optimizers directly. However, it is possible to continue with the use of a correlation-based optimization with modifications for tracking displacement throughout the sequence so that a time-accurate solution of \( \mathbf{\Gamma}(\mathbf{x}, t) \) is found. This is important for when estimates of velocity or acceleration are desired. For instance, a central-differencing scheme can be used for evaluating the velocity at time \( t_0 \) with an equal time separation between snapshots \( \Delta t = t_1 - t_0 \) (equivalent to a single-pair traditional PIV evaluation, i.e., Wereley, 2001). However, if the kernel includes additional times in the calculation, a higher-order scheme can be produced which reduces truncation error. For example, with \( \mathbf{\Gamma}(\mathbf{x}, t) \) available at four time instants, a fourth-order accurate scheme can be used. Thus from the perspective of truncation errors, developing a time-accurate estimate of the Lagrangian displacement field across a sequence of images can yield improvements in the final estimate of velocity at a single time instant. From the perspective of random errors, developing an estimate of the Lagrangian displacement field from multiple measurements may allow the influence of individual measurement errors to be reduced through, for example, least-squares fitting methods.

The concept of developing a time-accurate displacement estimate is similar to computer vision algorithms which perform a nonlinear tracking of features over multiple frames, such as Veeman et al. (2001) and Shaqique and Shah (2005). It is also similar to multiframe PTV algorithms using various motion models, such as Malik et al. (1993) and Li et al. (2008), among others. These works, however, consider the case of individual particle or feature tracking, which is known to suffer from reduced accuracy compared to the statistical cross-correlation used in traditional PIV (see e.g., Stanislas et al. 2008). The FTC algorithm uses cross-correlation to perform particle pattern tracking while also allowing for nonlinear motion. This combination of features is conjectured to provide greater accuracy and is further described here. The remainder of this section describes the algorithm, which consists of three steps: \textit{prediction} of the Lagrangian displacement field based on results obtained from traditional PIV algorithms, \textit{correction} of the predicted field using a correlation-based approach, and \textit{derivation} of the relevant quantities such as velocity and acceleration from the corrected field.

\textit{Displacement Field Prediction}

To establish a predictor the displacement field, the nomenclature of the processing must first be established. The FTC algorithm considers a kernel of \( N \) time-resolved PIV images \( I = \{I_{-n}, \ldots, I_n\} \) that correspond to snapshots of the flow acquired at times \( t = \{t_{-n}, \ldots, t_n\} \). The displacement field is then described as the trajectories of particle patterns over time, or equivalently, the position of interrogation grid points in time. These positions are denoted within the position vector \( \mathbf{X}(\mathbf{x}, t) \)
which is considered separately from $\Gamma(x,t)$; the former represents a discretized version of the displacement field, while the latter is a continuous representation which will be addressed in a following section.

The motion of particle patterns between individual pairs of images in this sequence can be determined using traditional PIV algorithms to yield a series of piecewise linear displacements $\mathbf{p} = \{p_{-n}, \cdots, p_{-1}, p_1, \cdots, p_n\}$. To summarize this nomenclature, figure 1 shows a graphical description of the quantities.

These predictors must be combined to form an initial predicted displacement field $X^0(x,t)$. One approach for performing this integration is to adopt a ‘pseudo-tracking’ approach as described by Liu and Katz (2006) in the context of material derivative estimation. This is a procedure where a sequence of velocity fields are used to propagate initial grid point positions in a piecewise manner forward and backward in time. Within this procedure, velocities at fractional grid point positions are determined using bilinear interpolation.

The approach used in this work is an extension of this technique, but based instead on a more advanced propagation procedure utilizing a fourth-order Runge-Kutta ordinary differential equation solver. Within the method, a bilinear interpolation is used for determining the velocity at fractional grid points and times. While a comparison is not presented in this paper out of need for brevity, this approach has been found to follow more accurately the true trajectories (as compared with synthetic data) particularly as the distance from the central image becomes larger.

**Displacement Field Correction**

The predicted field is now corrected such that the displacement field $X(x,t)$ represents the positions of particle patterns tracked across multiple frames as opposed to an integration of computed velocity fields. This is implemented within a predictor-corrector framework, where a corrector vector $X^k_c$ is generated at each iteration $k$. This is added to the predictor at iteration $k$ to form an updated predictor for the next iteration, i.e.,

$$X^{k+1}(x,t) = X^k(x,t) + X^k_c(x,t) \quad (4)$$

and graphically represented in figure 2.

To generate $X^k_c$, cross correlation is used between each image and the central image of the sequence, i.e., $I_0 \otimes I_n$, where $\otimes$ represents the normalized cross-correlation operator, and $I_n$ represents image $I_n$ which is deformed to $X^k(x,t_0)$. The position of the correlation peak indicates the correction needed to align the displacement field such that it represents the best match of the particle pattern across the sequence.

![Figure 1: Schematic of image and predictor nomenclature with respect to trajectory $\Gamma(x_0,t)$](image1.png)

![Figure 2: Schematic depiction of predictor-corrector strategy for one corrector iteration.](image2.png)
In the calculation of the corrector, the rotation of the trajectory between the images being correlated must be considered such that the domain of linear shifts represented in the correlation plane coincides with the local coordinates of the image window at the position where the corrector will be applied. This is achieved by calculating the angle between tangent vectors of the trajectory at the central time and the time of the image being considered. This angle is then used with a rotation matrix to rotate the corrector returned from correlation.

**Derived Quantity Estimation**

The corrected discrete displacement field \( \mathbf{X}(\mathbf{x}, t_0) \) can be considered as the best match of a particle pattern over the kernel of images, and now the estimation of the continuous displacement field \( \Gamma(\mathbf{x}, t) \) and its use in the calculation of derived quantities such as velocity and acceleration can be considered. The approach adopted by FTC is to describe the continuous displacement field using polynomial expressions which allow for a reduction in random errors from correlation via a least-squares fitting procedure, and also a reduction in truncation errors as outlined previously. In particular, first, second, and third order polynomials (\( P = 1, 2, 3 \)) are considered, which are independently applied to each coordinate direction. The polynomial approach is more general than other simple functions such as circular trajectories, and has a history of previous applications in the field of PTV for improvements in multiframe algorithms (second-order polynomials in particular, see e.g., Malik 1993, Li 2008) and Lagrangian derivative evaluation (Novara, 2012). As an example, the second-order polynomial approach results in an expression for the displacement field,

\[
\Gamma(\mathbf{x}, t) = a_0 + a_1 t + a_2 t^2
\]  

where the \( a \) coefficients are vectors defining the coefficient values for each grid point. The fitting procedure used herein first forces the constant term \( a_0 \) in the polynomial to the initial gridpoint of the central image so that the final result is returned on a regular grid, i.e., \( a_0 = \mathbf{X}(\mathbf{x}, t_0) \). This removes a parameter from the fitting procedure, and the remaining coefficients are then determined independently for each coordinate direction using a least squares fit found from the solution of the matrix equation,

\[
\mathbf{X}(\mathbf{x}) - \bar{\mathbf{X}}(\mathbf{x}) = \mathbf{M} \mathbf{a}
\]  

where \( \mathbf{M} \) is the design matrix whose form is dependent on the polynomial order chosen and the subtraction of the mean value \( \bar{\mathbf{X}}(\mathbf{x}) \) is used to enforce the fixed grid point constraint.

To make the estimate of velocity and acceleration, the analytical derivatives of the polynomials are used. This leads to a simple explanation of the resulting truncation errors; for example, a second-order polynomial used for describing \( \Gamma(\mathbf{x}, t) \) will yield a first-order description of \( d\Gamma(\mathbf{x}, t)/dt \) and a zeroth-order (constant) description of \( d^2\Gamma(\mathbf{x}, t)/dt^2 \). This has important implications for the dynamics which can be adequately modeled using a specific polynomial order; note that simply locally choosing the polynomial order such that the residual fitting error is minimized is not a valid approach since increasing the polynomial order will always lead to a reduction in residual error. The key is for the fitting to approximate the trend in the position, and not the individual values of the position, which are contaminated with cross-correlation error.

Using these polynomial expressions within a least-squares framework provides a unique ability for the FTC algorithm to transcend the inherent trade-offs typically made when using traditional PIV algorithms. In particular, these require that a value of \( \Delta t \) is optimized such that truncation errors are small, but the dynamic velocity range is large (see for example Prasad, 1992). FTC is able to operate on a different principle, where truncation errors can be minimized by considering multiple images. Additionally, the effective \( \Delta t \) of the trajectory is much larger and represents a
much lower relative error. These properties will be explored in the next section.

3. Assessment

The algorithm is now tested on sets of image sequence data and compared with results from a traditional single-pair PIV algorithm. Within this section two synthetic cases are evaluated: solid body rotation, which is used to estimate the reduction of truncation errors, and uniform flow, to estimate reductions in random error. Also a real case is used to demonstrate the capabilities of the method when confronted with real experimental data.

Solid-Body Rotation

Solid body rotation is a generalization of the wide variety of flows containing rotational motion. The particle displacement is radially-varying, with a tangential displacement given as $U_\theta = rU_0$, where $r$ is the radial distance from the centre of the image and $U_0$ is the rotation in radians between frames. This velocity distribution exhibits constant spatial gradients, allowing for a more direct comparison of the algorithms to be made without higher-order spatial deformations potentially affecting the results. The synthetic images are composed through integration of the intensity distribution of particles randomly distributed over an array of 500 x 500 pixels at a seeding density of 0.1 particles/pixel. The particle images are quantized by a mean intensity of 8 bits, and a mean diameter of 2.0 px. Out-of-plane motion and readout noise are not applied to the images. Additionally, no noise is added through a variation of intensity or diameter in order to specifically isolate the bias errors with minimal contribution from random error sources.

A series of 5 images were generated with a $U_0 = 10$ degree rotational step between each frame. For the interrogation procedure, the WIDIM algorithm (Scarano and Riethmuller, 2000) correlates the two images at the endpoints of the sequence (i.e., images 1 and 5), while FTC was applied to the entire sequence. The correlation settings chosen for WIDIM are two passes of 63 x 63 interrogation window size and 16 x 16 vector spacing, followed by three passes of 17 x 17 interrogation window size with 4 x 4 vector spacing. For both WIDIM and FTC, a Gaussian window function is applied to all correlations, and a sinc interpolation function with a 7 x 7 kernel is used for all image deformations. To cope with the large displacements that occur as the radius increases, an initial predictor is supplied to initialize the algorithm. For the application of FTC, the settings are identical to the final pass of WIDIM to allow for a direct comparison. Two iterations of the FTC algorithm are performed.

Figure 3 presents the bias error $\beta$ as a function of radial position for both methods. Error bars are also presented which correspond to the associated random errors $\sigma$ of the data. These errors are calculated as outlined in Astarita and Cardone (2005). Particularly, WIDIM displays a systematic overestimation of the velocity that increases with radial distance. In contrast, the FTC method has a behavior that depends on the polynomial order, but always returns an underestimation of the true velocity. However, the magnitude of underestimation by FTC is in each case lower than the magnitude of overestimation from WIDIM.

The overestimation by WIDIM is somewhat counterintuitive since truncation errors are typically associated with underestimates of behavior. Indeed, the effect of rotational motion on traditional algorithms has not been the target of detailed studies. However, the authors believe this discrepancy is due to the nature of the image deformation procedure, whereby interrogation windows at a particular radial position extract image data from positions which lie on a streamline of greater radius that is associated with a greater linear velocity. In contrast, the FTC algorithm exhibits the opposite behavior; as radial distance is increased, the algorithm returns a reduced velocity compared to the actual velocity field (particularly for the second-order fit). This is due to the use of polynomial expressions, which are not capable of matching exactly the dynamics of a solid body.
rotation over multiple frames. However, the magnitude of this reduction is much smaller than the overestimation of the WIDIM algorithm and represents a notable improvement.

Figure 3 makes clear why the polynomial order chosen should be high enough to approximate the dynamics of interest. For example, solid body rotation is a motion which contains a constant acceleration in time. This would suggest that the dynamics could be adequately modeled using a second-order polynomial, which is linear in velocity and constant in acceleration. However, when using second-order polynomials, a bias error is still present. This effect is because the constant acceleration of solid body rotation is with regard to the polar coordinate system, i.e., constant acceleration with respect to the center of rotation, while FTC considers polynomial fitting in Cartesian coordinates. The transformation between the two yields varying components of the acceleration based on the angle under consideration, therefore requiring a polynomial fit whose second derivative is at least linear in time.

Figure 3: Bias error comparisons between WIDIM and FTC algorithms for (left) single rotation angle of 10 deg, and (right) over a range of rotation angles, with the error magnitude normalized by radial distance. Error bars refer to standard deviation of error values. 5 images used in the FTC kernel.

To further demonstrate the reductions in truncation error, the analysis is extended by evaluating the effect of increasing solid body rotation angles (equivalent to increasing the ∆t of the measurement). It is expected that for increasing rotation angles, the growth of error in traditional algorithms will outpace that of FTC. Figure 3 shows this behavior for solid body rotations of up to 90 degrees. Notably, the same trends persist and the error growth of FTC is lower than WIDIM. Again, the third-order fit for FTC allows the algorithm to handle large rotations with minimal bias error as well as random error, while the second-order fit cannot adequately handle the dynamics and thus also results in a larger random error component as indicated by the error bars. Adding additional images to the FTC kernel was considered to evaluate any changes in trends. While not presented here, FTC was also applied using a 9 image kernel. The trends are nearly the same, indicating that the primary driver of truncation error reduction is the choice of polynomial order, and not the number of images used in the kernel.

Uniform Flow

A test based on uniform flow is devised to demonstrate reductions in peak locking bias error and random error due to the correlations performed at multiple effective ∆t in FTC, coupled with the least-squares fitting procedure. The displacements considered range from 0 to 2 pixels and are
generated using a large (2000 x 2000) image in which a very small velocity gradient is imposed to generate the range of velocities within a single image. The same parameters are used as previously for the generation of synthetic images; however, the particle diameter is reduced to a value of 1 pixel to accentuate the random errors and peak locking. The same interrogation procedures are also followed for both WIDIM and FTC; however, WIDIM is now used to correlate adjacent frames instead of the endpoints of the sequence.

![Graph](image1)

**Figure 4:** Error comparisons between WIDIM and FTC algorithms for uniform displacement from 0 to 2 pixels using a 5 image kernel for FTC. Left, bias error; Right, total error.

![Graph](image2)

**Figure 5:** Error comparisons between WIDIM and FTC algorithms for uniform displacement from 0 to 2 pixels using a 9 image kernel for FTC. Left, bias error; Right, total error.

Figure 4 shows the bias error as a function of displacement for WIDIM and FTC using a kernel of 5 images. FTC departs from the traditional algorithm in the behavior of the bias error; using a first or second-order fit does not exhibit the characteristic oscillation of 2 pixel wavelength, but oscillates at a greater frequency of 1 pixel. Additionally, the maxima of the oscillations are lower compared to the traditional algorithm. However, when using a third-order fit, the FTC algorithm behaves very similarly to the traditional algorithm and in some cases shows larger maxima in the error. Another perspective can be found in figure 4 which shows the total error. Here, the stiffness of the first and second-order fits provide a reduction in total errors up to an order of magnitude.
compared to WIDIM. This is remarkable because the order of magnitude decrease in error is accomplished using only 5 separate images. Even across the entire range of displacements, FTC returns a result with 3-5 times lower total error. When the third-order fit is used, FTC behaves similarly to WIDIM for a majority of displacements, but does display a downward trend in error at half-integer displacements.

The effect of increasing the number of images in the FTC kernel should have a much more pronounced effect in this case than in the previous case with solid body rotation. Figure 5 shows the same error plots as figure 4, but with 9 images used in the FTC kernel. Very clear improvements can now be seen between the two algorithms, with even the third order polynomial fit returning large reductions in total error. This can be further extended to evaluate the trends as the number of images are further increased. Figure 6 shows the variation in average total error as a function of the number of images in the FTC kernel. Also given are two trend lines to indicate scaling parameters. In general, the reductions in random error given by FTC scale by an amount greater than N; however, the initial reduction in error does not directly scale by this amount.

Submerged Turbulent Jet

The previous cases tested synthetic data where the true velocity field was known and error sources could be carefully controlled; now the focus turns to the applicability of the method for real data sets. In this section, the applicability of the FTC algorithm to TR-PIV data from a circular jet in water is considered. The flow issues upward from a 10 mm diameter nozzle and at a centerline velocity of 0.45 m/s into a tank of approximately 1 m diameter and 1 m height. The Reynolds number in accordance with nozzle diameter and exit velocity is 5,000. For seeding, 10 micron polyamide particles are introduced into the tank, which are continuously recirculated through the pumping system and the nozzle. A 1.5 mm thick laser sheet is formed from a Quantronix Darwin-Duo Nd:YLF laser (2 x 25 mJ/pulse), and is imaged using a pair of Lavision HighSpeedStar 6 CMOS cameras (1024 x 1024 px, 20 micron pixel pitch) at 1.2 kHz. This results in a maximum of 7 pixels displacement between subsequent frames occurring within the jet core. For more details on the configuration of this experiment, the reader is referred to the descriptions provided by Violato and Scarano (2011).

For the interrogation procedure, the previously used settings are applied. Figure 7 presents a comparison of the radial velocity component returned by the two methods which is a clear representation of the periodic vortex growth process occurring in the near field of the jet. This velocity component also is characterized by overall lower displacement magnitudes compared to the axial velocity; therefore, the relative error is greater and more easily visualized. This is illustrated in the top figure, where the relative error appears as a reduction in the coherence of the contours. Also, further in the progression of the jet the noise results in the distortion of the vortical structures. In comparison, the FTC algorithm returns a smoother result indicative of lower errors throughout the image. Note also that the shape and peak of vortical structures appears to be more clearly captured compared to the traditional PIV algorithms.
Figure 7: Instantaneous snapshot of the velocity normal to the jet axis. Comparison of WIDIM and FTC solutions for various polynomial orders and numbers of images used in the FTC kernel.

To quantify the random errors with more clarity, it is useful to consider a time-series signal generated by extracting a single grid point in each image and comparing the temporal coherence of
the signals for each of the algorithms. For an interrogation generating a greater amount of noise, large fluctuations will result when using differencing schemes to evaluate the temporal derivatives. Figure 8 presents a short time series of Eulerian acceleration calculated via central differencing from a point located in the shear layer region with periodic vortex growth. The autocorrelation of the signal is shown on the right; this provides a quantitative measure of the temporal coherence. A smooth reduction in the autocorrelation function indicates that variations in the considered variable are slowly varying in time, indicative of the propagation of structures in a fluid. In contrast, sharp reductions as seen with the single frame evaluation suggest that the predominant changes between time instants are due to noise fluctuations which are uncorrelated and do not vary slowly in time.

![Figure 8](image)

Figure 8: Subset of the time series at a single point in the PIV datasets, located in the shear layer region with periodic vortex growth. Left: Comparison of time traces; Right: Autocorrelation of time series data.

### 4. Conclusions

A novel algorithm for improving the processing of time-resolved image sequences has been presented and shown to reduce bias errors generated from both truncation effects and peak locking, as well as random errors, compared to traditional, single-pair processing algorithms. The algorithm has been found more capable than traditional PIV algorithms for tracking nonlinear motion in time due to the unique trajectory finding procedure and least-squares fitting. Additionally, the algorithm is able to use a larger number of images compared to other correlation averaging approaches since a linear predictor is not assumed across the sequence.

Regarding random errors, the achievable reductions are shown to be highly dependent on the choice of fitting function applied to the determined trajectory. However, with each polynomial order, the trend of the average error reduction was found to be greater than 1/N. The use of a second-order polynomial was found to have very similar error reduction properties to the first-order (linear) polynomial fits. Both methods achieve at least an order of magnitude reduction in average total error when using 10 or more images.

On the other hand, the separate mechanism leading to truncation errors can be reduced to a degree using a second-order polynomial, but even more effectively through the use of higher-order polynomials which allow for the fitting of more complex dynamical behavior, as shown in the synthetic case of solid body rotation. This indicates a possible application rule of the FTC algorithm in that vortical flows should be processed with third-order or higher polynomials. For regions of the flow with simpler dynamics (i.e., primarily advective motion), lower order polynomials can be used to provide more substantial error reductions.
References


