fluids

MDPI

Article

A Novel Polynomial Approach for Particle Image Velocimetry (PIV) Image Reconstruction

Briana M. Steven and Paul D. Docherty *

Department of Mechanical Engineering, University of Canterbury, Private bag 4800, Ilam, Christchurch 8041, New Zealand

* Correspondence: paul.docherty@canterbury.ac.nz; Tel.: +64-33692230

Abstract

Particle Image Velocimetry (PIV) often utilizes a cross-correlation method to determine how far particles have moved between two captured images. The most common methods for vector estimation use computationally exhaustive cross-correlation functions across the interrogation window and an exhaustive search to find the maximum correlation position. This paper proposes a novel method to vector generation in which a preprocessing blur is applied to the two image before performing a cross-correlation for only nine points. These nine points are used to approximate the original cross-correlation surface as a second-order polynomial surface that can be solved analytically to find the optima point. Three iterations of the process are used for each location converging to a precise optimum. This method is very accurate on computer-generated PIV images and solves the entire vector field faster than the original basic method at any image size. However, the success is limited to in silico PIV data and cannot produce coherent vector fields when applied to experimental data captured on a supra-aortic bypass PIV experiment. This method may find applications in other domains where the input data is closer to the perfect computer-generated particle data.

Keywords: particle image velocimetry; image reconstruction; numerical analysis; experimental fluids

Academic Editor: D. Andrew S. Rees

Received: 21 January 2026
Revised: 11 February 2026
Accepted: 13 February 2026
Published: 18 February 2026

1. Introduction

Particle Image Velocimetry (PIV) is a tool for analyzing flow fields in an experimental environment [1]. PIV typically seeds a working fluid with small particles that are illuminated by a plane laser [2]. A high-speed camera captures an image pair at a set time interval (Δt) and the displacement of the particles between the two images divided by Δt to produce a velocity field. The method of determining the displacement has developed over the years with advancements in digitization, computing power, and camera technology [3]. In particular, techniques for estimating the velocity field from the particle images have been developed steadily through the 1990s and 2000s [4]. However, there has been a paucity of recent developments in this field.

Popular current methods for PIV image analysis include image deformation [5,6], window shifting [7], interpolation schemes for sub-pixel accuracy [8], and the use of fast-Fourier-transforms [9]. All of these methods reach beyond the basic cross-correlation methods used in the early days of PIV [10–12]. Basic cross-correlation as described by Keane and Adrian [11] is shown in Equation (1). This method overlays small interrogation windows (I_1, I_2) from the two images and multiplies the pixel intensity values of the pixels

(from 0 to some maximum value) in corresponding positions (x, y), with the second image shifted by a displacement vector (u, v). The displacement vector is varied across the size of the interrogation window so all possible overlap positions are calculated. This results in a field of C values where the highest value of C denotes the displacement vector with the best match of the two window's pixel intensity values.

$$C = \sum I_1(x,y) \cdot I_2(x+u, y+v) \tag{1}$$

Image-deformation methods introduce a second step after an initial run of basic cross-correlation across an interrogation window. This is an image-resampling step which determines a new layout of points corresponding to the (x, y) positions in the undeformed interrogation window using Equation (2). All terms on the right-hand side come from the data collected from the initial basic cross-correlation.

$$\begin{aligned} x_s &= x + \left(\frac{\partial u}{\partial x}x + \frac{\partial u}{\partial y}x\right) \\ y_s &= y + \left(\frac{\partial v}{\partial x}y + \frac{\partial v}{\partial y}y\right) \end{aligned} \tag{2}$$

As the resulting (x_s, y_s) is often sub-integer, an interpolation scheme is needed to determine what the corresponding pixel-intensity value should be based on the original I_2. This sub-pixel interpolation can be performed with different interpolation schemes, for example bilinear [6], B-spline [13] or Gaussian peak interpolation [8]. Sub-pixel interpolation can also be used directly after an initial run of basic cross-correlation. Once the displacement vector is found at an integer whole-pixel level, iterative rounds of sub-pixel shifts in the interrogation window on the second image are performed until the difference to the previous round is close to zero.

Window shifting involves the application of a central difference interrogation algorithm, rather than the forward one employed by conventional PIV analysis techniques. Interrogation windows are shifted $-V(x,y)\Delta t/2$ and $V(x,y)\Delta t/2$ from a starting grid of points at time t, midway between the two image captures [7]. Given this process involves knowing the displacement of the interrogation window an iterative method must be involved using an initial run of basic cross-correlation.

The basis of these more complex methods still rest on the fundamental step of calculating a cross-correlation field for individual interrogation areas, which can lead to lengthy calculation time that grows with the size of the interrogation window. Some methods, such as the window shifting and sub-pixel interpolation also involve unknown rounds of iteration. This further adds to the calculation load and time. Another issue when using methods involving an interpolation scheme is peak-locking [14]. Peak-locking error comes from the sub-pixel interpolation schemes being used on originally discrete distributions from the cross-correlation step.

This paper proposes a polynomial approximation method for PIV image reconstruction. By employing a preprocessing blur on the image pairs, the cross-correlation field becomes rounded and the surface can be approximated by a standard quadric equation [15]. While other image-analysis methods use interpolation schemes to find the sub-pixel location for the best fit displacement, this polynomial method has an analytical solution for the best-fit vector so no interpolation scheme is needed.

2. Materials and Methods

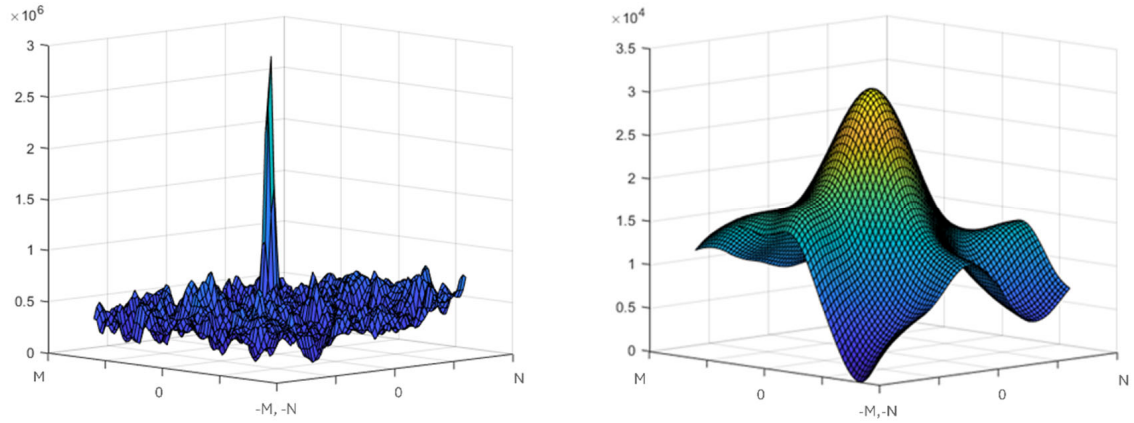
2.1. The Proposed Polynomial Solution Method

The two images $I_{1,0}$ and $I_{2,0}$ from the collected image pair are blurred using a 2-D Gaussian smoothing kernel with a standard deviation of 8 and 0 edge padding to give

I_1 and I_2. This relaxation of the objective surface yields a smooth objective surface that can be relatively well defined by hyperbolic expressions. Figure 1 shows the standard objective surface often used in estimation of a velocity vector ($[u,\ v]^T$) at some point ($[x,\ y]^T$) (left) compared to surface generated by the proposed blurring method (right). To ensure consistency with published methods in PIV analysis, both of these surfaces were generated with the cross-correlation equation (Equation (3)) rather than the objective surface equation of the proposed method.

$$\psi(u,v) = \sum_{m=-M}^{M} \sum_{n=-N}^{N} I_1(x+m, y+n) * I_2(x+m+u, y+n+v) \tag{3}$$

where $\psi(u,v)$ is the cross-correlation value for a given vector displacement (u,v). $I(x + m, y + n)$ is the light intensity value for the given pixel at $(x + m, y + n)$ in the first image. The domain of consideration is centered at x, y and ranges across $2M + 1$ by $2N + 1$; I_2 is the intensity at the displaced second image. The candidate values of u and v range are evaluated on a domain from $-M$ to M and $-N$ to N.



(**a**) (**b**)

Figure 1. Comparison of the objective surfaces of the traditional method (**a**) and the proposed method (**b**). To aid interpretability, the figures showing the smoothed correlation surface have been flipped to demonstrate the method clearly. The yellow color indicates a higher correlation value on the surface.

The proposed method uses a different approach the $(2M + 1) * (2N + 1)$ grid search for each location of interest. A grid of nine points is located around the best u and v estimates (u', v', respectively) on the interrogation window, defined by Equation (4). Initially, u' and v' were defined as zero, and the half-range of the initial grid (ϕ_b) is defined as five pixels. ϕ_b determines the size of the domain of nine candidate vectors that are used to characterize the objective surface across. The refining of ϕ_b in successive iterations allows the algorithm to converge upon an optimal vector.

$$\begin{aligned} \mathbf{g} &= [g_1, g_2, g_3] = [-\phi_b, 0, \phi_b] + u_i' \\ \mathbf{h} &= [h_1, h_2, h_3] = [-\phi_b, 0, \phi_b] + v_i' \end{aligned} \tag{4}$$

The u and v values of those nine points are used in Equation (7) nine times to generate the matrices defined in Equation (6). The alpha values in Equation (7) are unknowns, and

will define the shape of the polynomial surface generated by Equation (5) that best matches the correlation surface created with Equation (5)

$$\psi(g_k, h_l) = \sum_{m=-M}^{M} \sum_{n=-N}^{N} \left(I_1(x+m, y+n) - I_2(x+m+g_k, y+n+h_l)\right)^2$$
$$\psi(g_k, h_l) = \alpha_1 g_k^2 + \alpha_2 h_l^2 + \alpha_3 g_k + \alpha_4 h_l + \alpha_5 g_k h_l + \alpha_6 \tag{5}$$

$$\mathbf{A}\boldsymbol{\alpha} = \boldsymbol{\psi}$$

where

$$\mathbf{A} = \begin{bmatrix} g_1^2 & h_1^2 & g_1 & h_1 & g_1h_1 & 1 \\ g_1^2 & h_2^2 & g_1 & h_2 & g_1h_2 & 1 \\ g_1^2 & h_3^2 & g_1 & h_3 & g_1h_3 & 1 \\ g_2^2 & h_1^2 & g_2 & h_1 & g_2h_1 & 1 \\ g_2^2 & h_2^2 & g_2 & h_2 & g_2h_2 & 1 \\ g_2^2 & h_3^2 & g_2 & h_3 & g_2h_3 & 1 \\ g_3^2 & h_1^2 & g_3 & h_1 & g_3h_1 & 1 \\ g_3^2 & h_2^2 & g_3 & h_2 & g_3h_2 & 1 \\ g_3^2 & h_3^2 & g_3 & h_3 & g_3h_3 & 1 \end{bmatrix}; \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{bmatrix}; \boldsymbol{\psi} = \begin{bmatrix} \psi(g_1, h_1) \\ \psi(g_1, h_2) \\ \psi(g_1, h_3) \\ \psi(g_2, h_1) \\ \psi(g_2, h_2) \\ \psi(g_2, h_3) \\ \psi(g_3, h_1) \\ \psi(g_3, h_2) \\ \psi(g_3, h_3) \end{bmatrix} \tag{6}$$

$$\boldsymbol{\alpha} = \mathbf{A}^{-1}\boldsymbol{\psi} \tag{7}$$

Those alpha values are evaluated with Equation (7) using the inverse operation (in this paper, the MATLAB 2022b inverse operator was used). Once Equation (5) is fully defined, the minima of the surface is found with the first derivatives of the surface, Equation (8). The derivatives in Equation (8) can be set to zero to locate the minima, Equation (8). Solving the two resulting equation $\left(\frac{d\psi}{du}, \frac{d\psi}{dv}\right)$, Equation (8) gives the optima u and v locations ($\left[u_{opt}, v_{opt}\right]^T$).

$$\frac{d\psi}{du} = 2\alpha_1 u_i + \alpha_3 + \alpha_5 v_i$$
$$\frac{d\psi}{dv} = 2\alpha_2 v_i + \alpha_4 + \alpha_5 u_i$$
$$\mathbf{P}\left[u_{opt}, v_{opt}\right]^T = \mathbf{q}$$

where

$$\mathbf{P} = \begin{bmatrix} 2\alpha_1 & \alpha_5 \\ \alpha_5 & 2\alpha_2 \end{bmatrix}; \mathbf{q} = \begin{bmatrix} \alpha_3 \\ \alpha_4 \end{bmatrix} \tag{8}$$
$$\left[u_{opt}, v_{opt}\right]^T = \mathbf{P}^{-1}\mathbf{q}$$

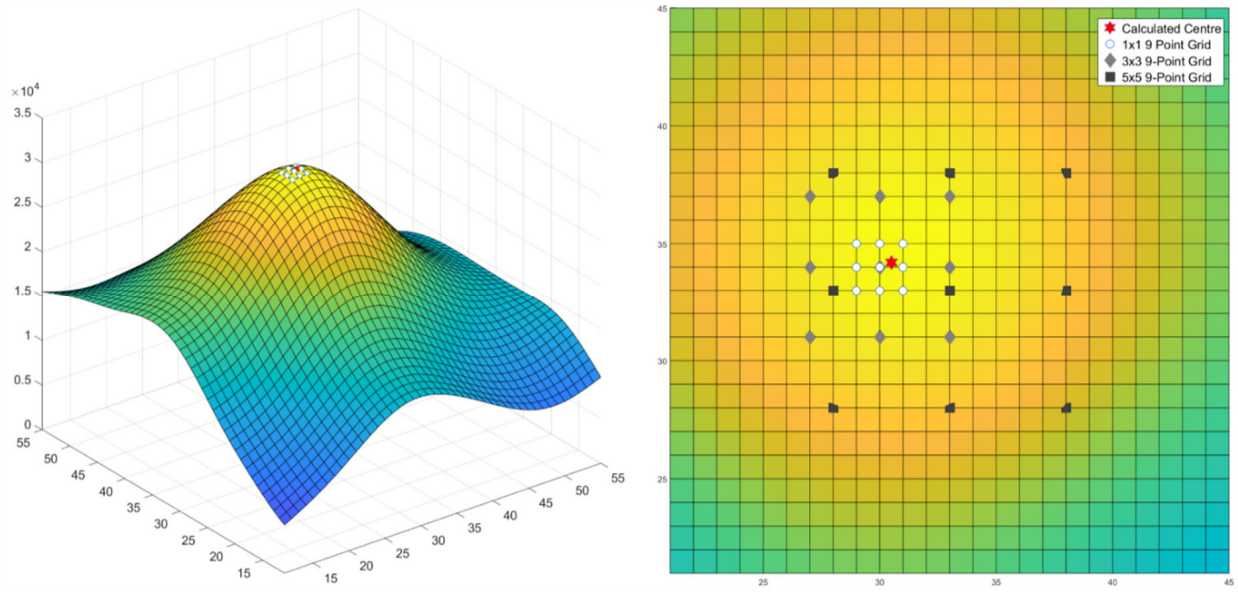
As the original array of light intensity values is discretized, Equation (7) finds the closest pixel location for the minima point, and the process is repeated from Equation (4) through Equation (7) where $\phi_b = 3$, and a final time from Equation (4) through Equation (8) where $\phi_b = 1$ ($\phi_b \in \{\ \}$).

Equations (3)–(9) are repeated across $i = \{1, 2, 3\}$ iterations with $\phi_b = 5,\ 3,\ 1$, respectively. For the final iteration ($\phi_b = 1$), Equation (9) is ignored. The locations of interest are shown for an arbitrary case in Figure 2b.

$$u'_{i+1} = u'_i + round(u_{opt}),\ v'_{i+1} = v'_i + round(v_{opt}) \tag{9}$$

The final u_{opt} and v_{opt} values (depicted as the red star in Figure 2) provide the optimal shift between the two images indicating the velocity vector. The process is repeated for every location of interest to build a velocity vector field. The interrogation areas used in this implementation were 65 × 65 pixels ($M = N = 32$). The velocity field was identified at 12 pixel increments to generate a comprehensive velocity vector field. To avoid issues

with incomplete interrogation fields, the first location of interest was located 38 pixels from the edge.



(**a**) (**b**)

Figure 2. Correlation surface after blurring (**a**) with grid points used in polynomial surface equation pictured, (**b**) is a zoom in of the central area of the surface. The yellow color indicates a higher correlation value on the surface.

2.2. Validation of Solution on Synthetic Data

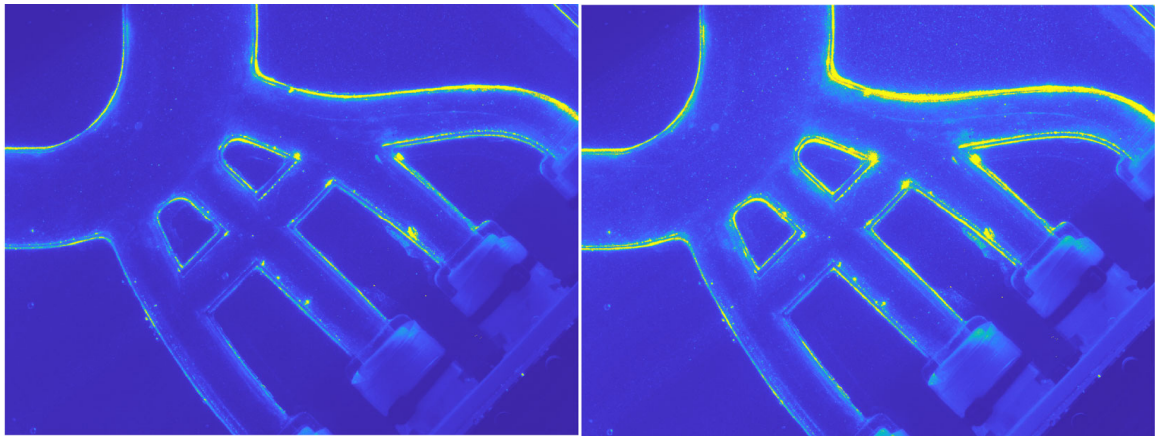
The initial validation of this method uses reference velocity vectors created with the in silico PIV-image-generation code developed by Mendes et al. [16]. The generation code outputs two PIV images based on inputs such as flow type, particle size, particle density, and velocity. The proposed method was tested with the flow type 'Rotated Shear Flow' which demonstrates a range of velocity magnitudes, and areas of flow that required sub-pixel accuracy to resolve. The inputs to the image-generation code are as follows: Pixel Radius = 3.0; Particle Density (Ni) = 12; Displacement Ratio (DeltaXFactor) = 0.2; Noise Level = 0; Out-of-plane standard deviation = 0.025. The velocity vectors provide a true solution to compare the results of the proposed method against. To compare the generated vectors to the reference vectors the average error per vector was found for several different particle sizes, maximum speeds and particle density. This average error per vector was taken from the total error summed from every comparison between the magnitudes of the generated and reference vectors as shown in Equation (10). The maximum relative vector error was also compared to the largest vector magnitude present in the reference vectors.

$$\varepsilon = \left|\mathbf{u}_{reference,} - \mathbf{u}_{generated}\right| + \left|\mathbf{v}_{reference,} - \mathbf{v}_{generated}\right| \quad (10)$$

2.3. Validation of Solution on In Vitro Data

This method will also be validated on PIV image data from an experiment considering perioperative bypasses in the supra-aortic branches by Williamson et al. [17]. This experiment measured the flow in a compliant phantom of the aorta, supra-aortic arteries and their bypasses and pulsatile flow waves. The camera focus was set precisely to ensure the illuminated light sheet was in focus, and therefore there was minimal blur of the particles on the raw images. The raw image is of 1768 × 2352 pixel resolution, which allows for particles to show as an area of multiple pixels. The blur on the edge of each particle is

very minimal before the preprocessing blur is applied in the polynomial method. The images used in this paper were from the peak flow of this waveform. This peak flow point was selected, as there would be strong flow signals useful for intuitive interpretation and statistical analysis. The image pair is displayed in Figure 3, with a time difference of 400 microseconds between the two image captures. Note that light refraction from the laser caused bright spots on the phantom walls.



(**a**) (**b**)

Figure 3. Image pair from experimental data: (**a**) First image captured, (**b**) Second image captured (color scale yellow = highest light intensity, light intensity valves have been enhanced for display only to show the particles within the phantom more clearly).

Equations (1)–(7) were used without alteration across the in silico and in vitro datasets. Reference vectors were not available from the in vitro original experiment; however, they can be calculated using the cross-correlation method outlined above.

2.4. Processing Time Comparison Between Methods

The computer calculation time for a variety of image sizes was evaluated on an 11th Gen Intel® Core™ i7-11700 @ 2.50 GHz with 32 GB installed RAM (HP Inc., Palo Alto, CA, USA) using Matlab 2022b. The time taken to perform the vector analysis for each method was measured using the MATLAB 'tic-toc' function. The difference in time between the Basic PIV method and the proposed polynomial method can be calculated and normalized by the image pixel counts that were calculated.

3. Results for the In Silico Images

3.1. Single Example Result

Figure 4 shows the initial images and the resulting vector fields from both the reference and solution methods for the Rotated Shear flow. Figure 5a shows the error magnitude. The error across $y \in \{74, 134, 254\}$ values is plotted in Figure 5b showing the solution vectors v component closely matches the dashed reference v component apart from the peaks of the green line and the right end of the blue line. The first row of data in Table 1 shows the summary statistics for this flow type.

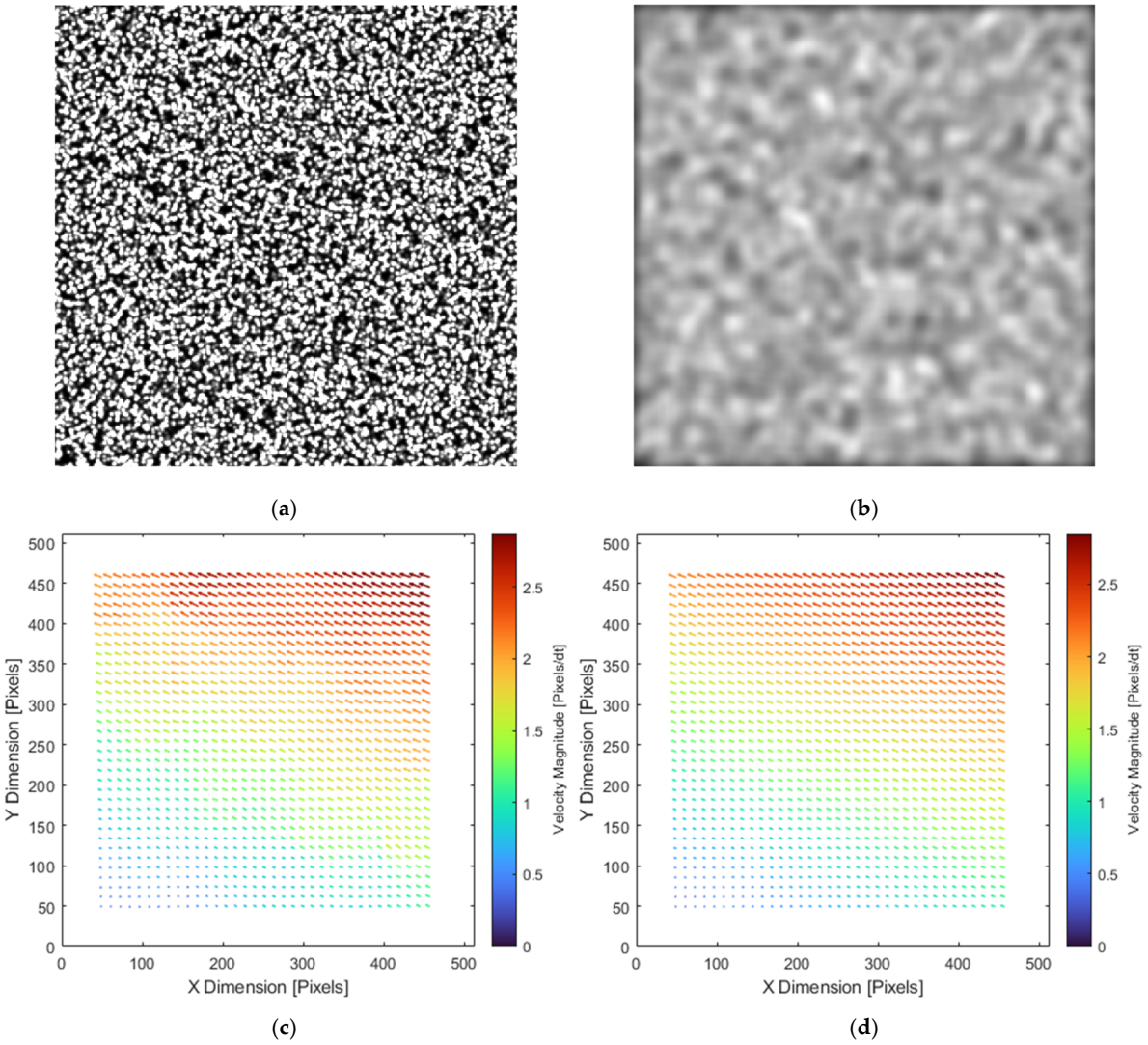


Figure 4. (**a**) Unblurred computer-generated particle image. (**b**) Blurred computer-generated image. (**c**) Vectors generated by the proposed method. (**d**) Reference vectors from the flow equation.

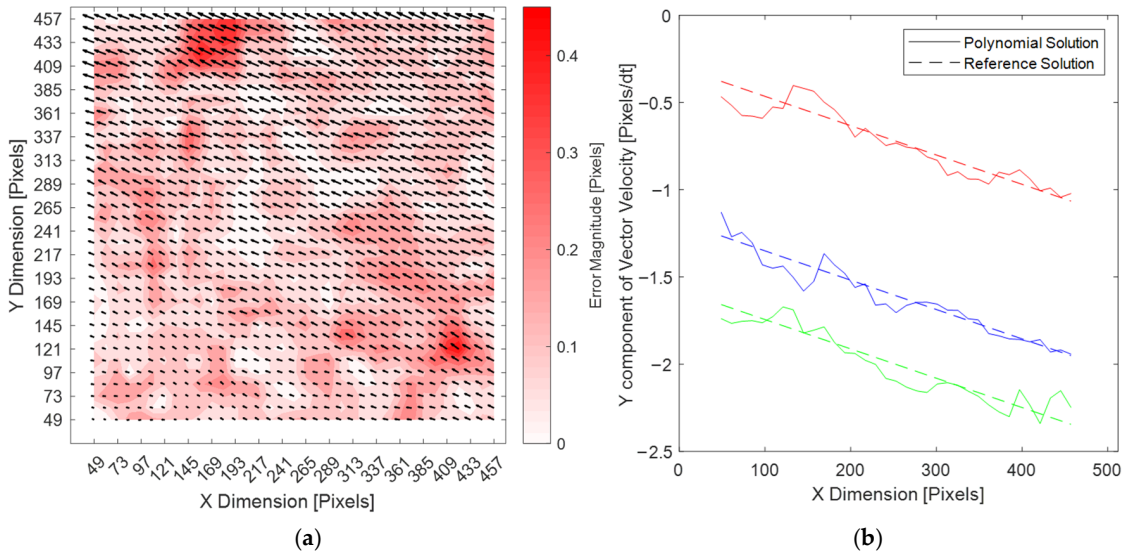


Figure 5. (**a**) Vector error at each vector position. (**b**) Solution and reference vector values for v component at certain y values (Green: Y = 385, Blue: Y = 289, Red: Y = 73).

Table 1. Error results.

In Silico Settings				Approach Assessment	
Flow Type	**Pixel Radius [Pixels]**	**Pixel Density [Particles Per Interrogation Area]**	**Maximum Velocity in Reference [Pixel/dt]**	**Relative Maximum Error to Max Velocity**	**Average Error Per Vector [Pixel/dt]**
Uniform Rankine	5.0	6	6.33	14.08%	0.208
“	2.0	5	3.17	19.20%	0.149
“	6.0	5	3.17	19.23%	0.180
“	2.0	10	3.17	17.17%	0.138
“	6.0	10	3.17	18.34%	0.160
“	2.0	5	6.33	15.29%	0.205
“	6.0	5	6.33	13.50%	0.214
“	2.0	10	6.33	19.68%	0.200
“	6.0	10	6.33	15.98%	0.241
Decaying Vortex	1.5	12	4.80	14.03%	0.209
Stagnation	4.0	8	5.78	8.18%	0.164
Parabolic	3.0	8	4.00	10.42%	0.122
Rotated Shear	5.0	5	5.87	9.14%	0.140

3.2. Error over Multiple Examples

Using the error measurements from Equation (8), the solution method was performed on a range of particle radius and particle densities, and a selection of different flow types. Table 1 displays the input variables and error results, where the first line is from the image pair analyzed in Figures 4 and 5.

3.3. Processing Speed Comparison Between the Original and Proposed Method

These results were found using the two methods performed on the Uniform Rankine flow type with pixel radius 5.0 and a Pixel Density of 6. The Speed Ratio value increases as the original image size increases. This demonstrates the difference in time between the Basic and Polynomial Proposed methods does not increase linearly with respect to the image size increase.

4. In Vitro Image Results

The blurred image is shown in Figure 6, with comparison to the unblurred image, and a zoomed in area of the flow path below. The code was run, and produced these vector fields in Figure 7. For clarity, Figure 7b shows the same outcome with all vectors over 15 pixels in magnitude removed.

Additionally, the code was run on the isolated section shown in Figure 6d to determine how it performs when not facing any effect from the wall brightness. The results of this, compared to the original cross-correlation method are shown in Figure 8.

It is clear the proposed approach failed to capture an accurate velocity vector field in the in vitro dataset. The coherence of the flow indicated by the traditional method [17] indicates that the problem was not with the data itself. Rather, the proposed method failed to properly identify the vectors from the data. The incoherence of the resultant vectors precludes further statistical analysis.

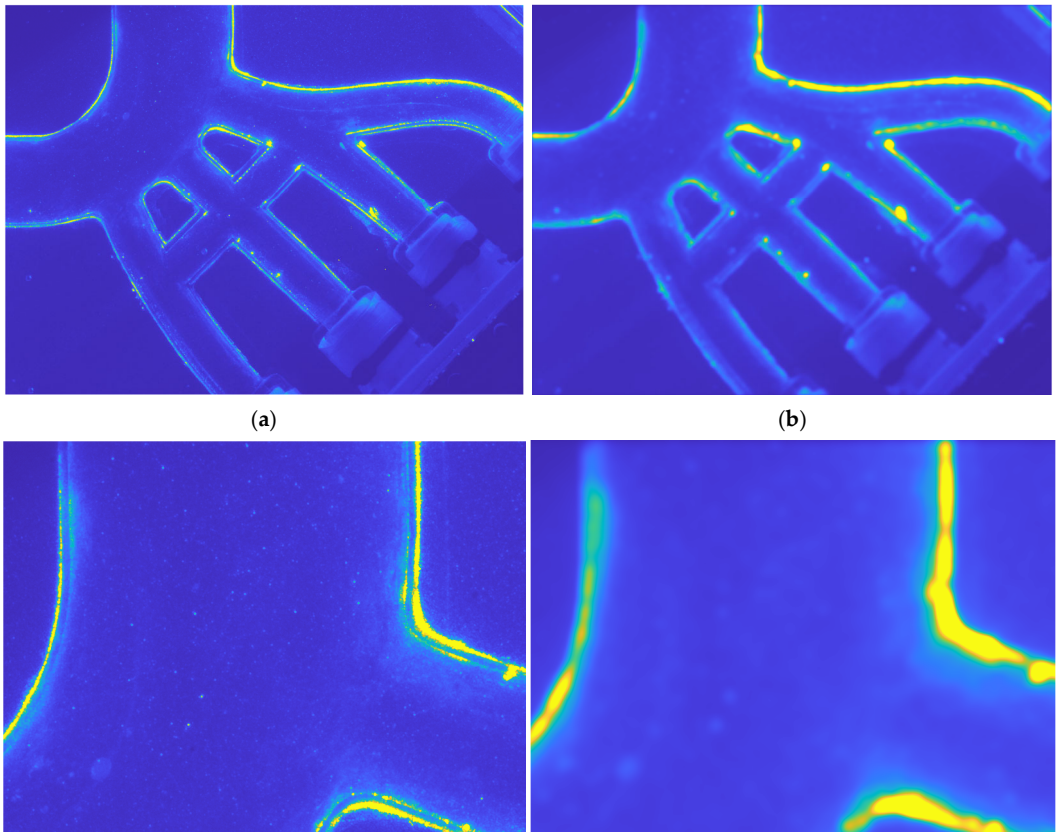


Figure 6. In Vitro images pre- and post-blur (**a**) Particle image of aortic geometry with efferent arteries and bypasses. (**b**) A blurred version of (**a**). (**c**) A zoomed in region of (**a**). (**d**) A blurred and zoomed in version of (**b**) (color scale yellow = highest light intensity, light intensity valves have been enhanced for display only to show the particles within the phantom more clearly).

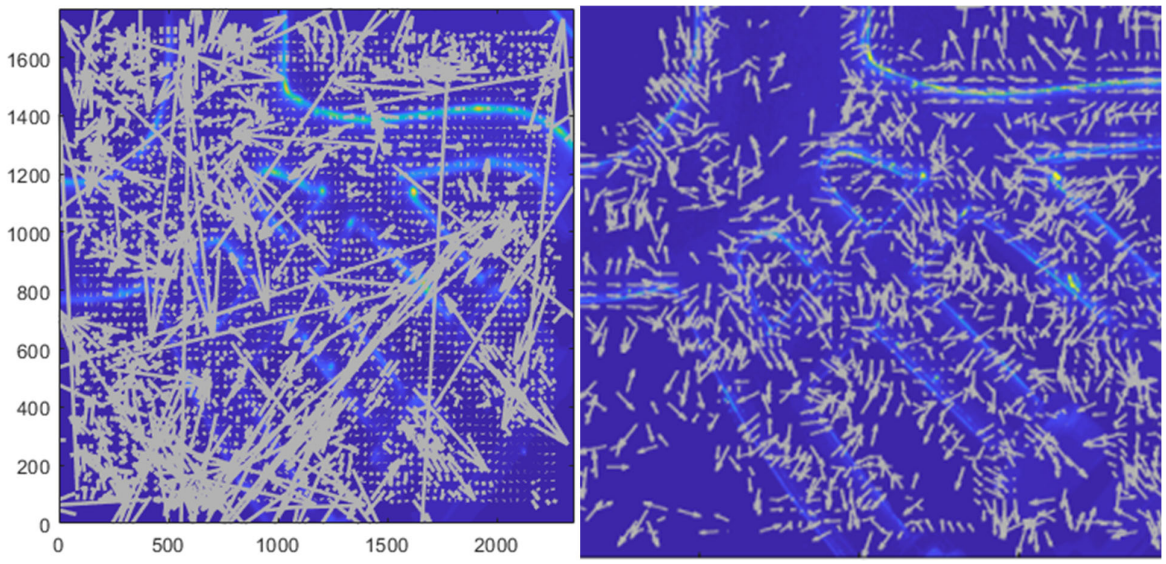
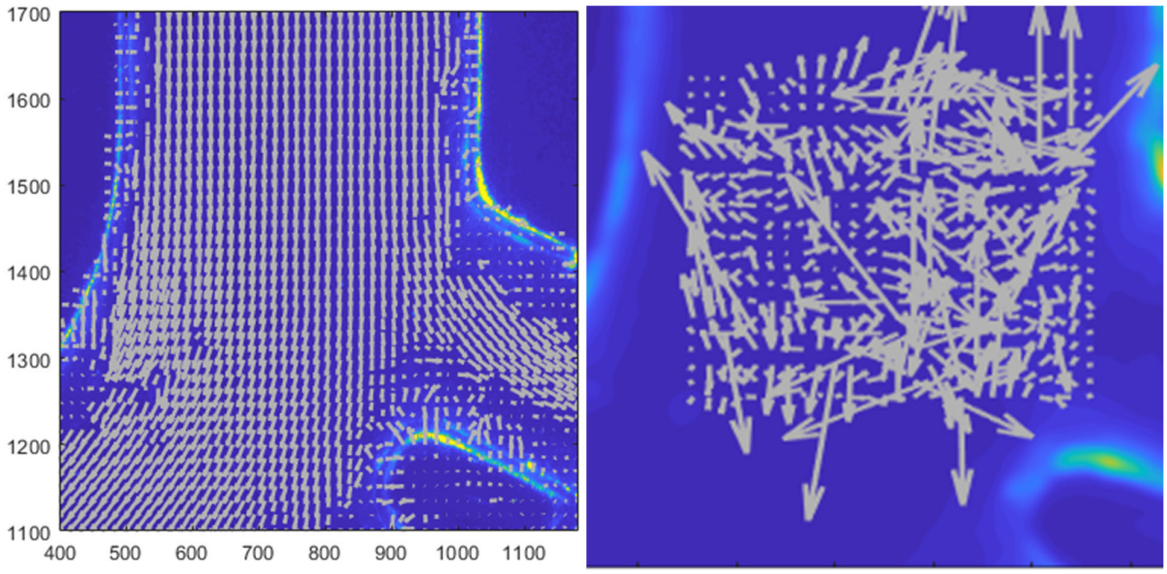


Figure 7. Vector Field results. (**a**) Unlimited vector length obscured by large vectors. (**b**) The same vectors limited to a magnitude of 15 pixels (color scale yellow = highest light intensity, light intensity valves have not been artificially adjusted; instead they are shown as used in the polynomial method).



(**a**) (**b**)

Figure 8. Zoom-in of vector field results from (**a**) the typical approach and data presented by Williamson et al. 2024 [17]. (**b**) The same data with the proposed method (color scale yellow = highest light intensity, light intensity valves have not been artificially adjusted, instead they are shown as used in the polynomial method).

5. Discussion of In Silico Data Results

The in silico results (Figures 4 and 5, Table 1) indicated the proposed method was promising. The velocity vector fields generated by the polynomial approximation method look very similar to the reference velocity fields (Figure 4c,d). There was a low level of error in the vector field in a variety of flow cases (Table 1). For context, competitors in the PIV challenge reported errors in the range of 2–6 pixel magnitudes for a similar problem using a Rankine Vortex flow [18]. The proposed approach was robust to changes in the image generation code as the particle size and density was varied. This demonstrated the method's potential across a wide range of PIV situations. The maximum error percentage stayed under 20% for every case displayed in Table 1 and under an average of 0.25 pixel/dt per vector.

This low average error rate shows an improvement over methods that do not determine the sub-pixel cross-correlation peak. Methods that do not optimize to the subpixel scale have a minimum average expected error of 0.383 (based on $\int_{-0.5}^{0.5}\int_{-0.5}^{0.5}\|x,y\|_2 dxdy$). The lowest error for the proposed method occurred in the parabolic test case with small particles, moderate density and lower maximum velocity with an average error per vector of only 0.122 pixels. The accuracy which exceeds the theoretical maximum accuracy is due to the smoothing effects of the approach effectively capturing flow-vector coherence information from not just the optimal vector, but also from vectors close to the optimal vector. Furthermore, the appearance of PIV particles in discrete pixel-space does not necessarily preserve the pixel shape for a particle across image pairs. Hence, the relaxation (or convolution) phase of the proposed approach ultimately leads to a smoother objective surface that is more coherently observed by the optimization process.

The particularly strong performance in the 'parabolic' case was somewhat expected, as the parabolic flow type only has horizontal flow that changes magnitude depending on the y value. The solution with the lowest relative error was the 'stagnation flow' case with medium particle size, moderate density, and high maximum velocity (however, this had a higher average error per vector than the parabolic flow). Both of these cases had flow in only one orthogonal direction. When the particles are moving uniformly in one

direction, the vector has only one component, so the potential for error is reduced. The largest relative error occurred in the Uniform Rankine flow with small particles, high density, and high maximum velocity. This error behavior was expected, as a Rankine vortex has a sharp change in flow velocity at the horizon radius, which physical flow situations cannot experience due to the coherence between fluid layers.

The worst errors were in the velocity transition zones near the Rankine vortex horizon. The in silico simulation of the Rankine vortex had a sharp change in velocity profile, which is not typical in subsonic flows. Thus, any peak within the interrogation window is ultimately smoothed off. However, this error is common to most PIV approaches as velocity is typically assumed constant across the interrogation window. Nonetheless, increased camera resolution and a higher seeding-particle density would allow a smaller interrogation region and thus smaller averaging error.

The proposed method only needs to calculate 27 correlation values to find the optimum displacement position. This is significantly less than the typical $(2M + 1) * (2N + 1)$ exhaustive search. The results from Table 2 with the processing time calculations show this difference in computational burden has a demonstrable effect. Other methods, including those used by OpenPIV and PIVlab, perform an entire field cross-correlation before using a Gaussian-fit method to get the subpixel accuracy. This approach is computationally expensive for large interrogation windows, as the number of entries needing to be calculated in the correlation matrix is a 4:1 ratio of the number of pixels in the interrogation window. The proposed method bypasses the unnecessary calculations from $\{u, v\}$ values, which, after some preliminary calculations, are highly unlikely to contain the optima. By measuring the performance on the course 3×3 grid of nine points, then refining the grid size and reevaluating, the proposed approach is similar to a bisection method. This approach is only achievable due to the Gaussian smoothing of the original pictures. This smoothing allows locations proximal to the optima to have a moderate correlation score. This allows convergence to the optimal value. In contrast, when only points in space are illuminated, locations proximal to the optimum will have a low correlation, and an exhaustive search is necessary. When the image size increased, the speed ratio also increased. This indicates that the proposed polynomial method gets faster in comparison to the basic PIV method [19] when the image size is increased, and is still faster in the smaller images.

Table 2. Processing Time Comparison.

Image Dimensions (Pixels)	Basic PIV Method Speed [s]	Proposed Polynomial Method Speed [s]	Difference in Time (Δt) [s]	Total Pixels in Image	Speed Ratio $\frac{\Delta t}{M*N}$
512×512	0.912	0.740	0.172	0.262×10^6	0.656×10^{-6}
1766×2352	22.3	17.0	5.30	4.15×10^6	1.27×10^{-6}
7072×9408	494	281	213	66.5×10^6	3.20×10^{-6}

The validation process used when evaluating this proposed polynomial approach also has some limitations. Some of the aspects of in vitro data that were not included in this experiment were background noise from the material behind the experiment phantom, difference in contrast between images due to the laser pulse timing, out-of-plane motion or artifacts from either the camera, or flaws in the experiment materials. None of this is captured in the in silico data. Thus, the proposed approach was not validated with respect to these known effects. Nonetheless, this preliminary in silico analysis was undertaken to evaluate the approach with a known true value to compare the optimization against. This direct comparison allowed a direct validation against known values rather than the axiomatic validation of using in vitro data. The images were generated from exact

equations, and thus the reference vectors could be found directly from these equations. The majority PIV work is applied to in vitro flows. However, the accuracy of a proposed method is much harder to evaluate statistically when applied to data that lacks a known perfect solution. The behavior of the in vitro flow can be theorized based on the experiment set-up and previous published work on the flow characteristics but there will not be a perfect reference to evaluate against.

6. Discussion of In Vitro Results

The proposed method was proven to be efficient and effective when applied to PIV data that was generated in silico. However, the proposed method yielded unusable, incoherent images when applied to the in vitro PIV images (Figure 7). The traditional method was successful in the in vitro data (Williamson et al. [17]) and exhibited coherent image reconstruction using the same data (Figure 7). A large portion of the vectors in the in silico data are above 15 pixels in magnitude, which is equivalent to 2.25 ms^{-1}. The known velocity was in the region of 0.1 to 0.5 ms^{-1}, which is shown in the findings of Williamson et al. [17]. That velocity equates to a pixel offset of approximately one to four pixels between image pairs, and was well within the accepted recommendation of pixel offsets no more than 1/4th the size of the interrogation window [20] of 32 pixels by 32 pixels. This pixel offset would have been within the $-5 \rightarrow +5$ pixel bounds of the proposed methods. The consistently erratic vector directions and magnitudes indicate an absolute failure in the method. As the haemodynamics this in vitro experiment is mimicking are through cylindrical smooth geometry with simple boundary conditions, the flow vector field is expected to be coherent with no discontinuities. This is because the flow must adhere to the Navier–Stokes equations [21], and at the Reynolds Number calculated for this in vitro example [17] the flow is expected to be laminar [22]. The incoherence of the flow profiles generated by the proposed method imply that statistical analysis is superfluous.

As Figure 6 shows, the wall imperfections dominate, as they are much brighter than the areas around them, and the zoom in section in (Figure 6c) displays how much brighter the wall is than the particles in the flow space. After blurring, these particles lose light intensity and noise in the background is amplified. While this issue could be mitigated by masking the wall regions, incoherent vector estimations were observed in regions distant from the walls (Figures 7 and 8). These results mean the approach, which was validated in in silico data, was not valid for use in in vitro data.

To attempt to overcome the issues with the application of the method to the real data, a number of preprocessing methods were applied to the in vitro data images. These included applying scaling curves to ensure light-intensity matching across image pairs and transforming the images using sigmoidal functions that amplify particles and mitigate the lightness where there were no pixels apparent, and thresholding. However, none of the preprocessing methods yielded suitable velocity vectors.

While the in vitro results were disappointing and indicate the approach is not currently fit for our purpose, there is excellent performance of the approach in in silico data as well as apparent, and theoretically justifiable, consistent improvement of the theoretical limit of precision. The authors feel this implies that it is suitable for presentation to the academic community. Despite the initial success of the in silico results it was important to consider in vitro testing. PIV is a useful tool for validating computational fluid dynamic models or to capture flow with complex fluid to solid interactions. This is why testing and reporting on the proposed approach with PIV data was important.

Future work to improve this method for use in PIV experimental image processing seems possible if the right prepossessing is determined. However, such an augmentation

will alter the in vitro image characteristics to conform to the characteristics of the in silico data and therefore could negate the savings in time and computational load.

7. Conclusions

While there has been stagnation in the development of new PIV image analysis methods in recent decades, the proposed approach described in this paper shows there is still potential to progress the field. To the authors' knowledge, the proposed approach is a novel concept that utilizes image blurring to allow smoother objective surfaces that conform to second-order parabolic functions local to the optima point, especially in in silico-generated data. The low relative errors over a large variety of flow types, particle sizes, and densities demonstrate the flexibility and precision of the method. However, the proposed method was unable to find a coherent vector field for the in vitro images. Nonetheless, the proposed method is novel and solved a particular problem very quickly and with high accuracy. Hence, it may be suitable for a class of image analysis tasks outside the PIV field.

Author Contributions: Conceptualization, P.D.D.; methodology, B.M.S. and P.D.D.; software, B.M.S. and P.D.D.; validation, B.M.S. and P.D.D.; formal analysis, B.M.S. and P.D.D.; investigation, B.M.S. and P.D.D.; resources, B.M.S. and P.D.D.; data curation, B.M.S.; writing—original draft preparation, B.M.S.; writing—review and editing, B.M.S. and P.D.D.; visualization, B.M.S.; supervision P.D.D.; project administration, P.D.D.; funding acquisition, B.M.S. and P.D.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the University of Canterbury Aho Hīnātore accelerator PhD scholarship.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data for the in silico analysis can be recreated using the code provided in Mendes et al. [16]. The in vitro data was shared with us, and we cannot disseminate it further.

Acknowledgments: We thank Petra Williamson for sharing the in vitro data with us.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

PIV Particle Image Velocimetry

References

1. Adrian, R. Twenty Years of Particle Image Velocimetry. *Exp. Fluids* **2005**, *39*, 159–169. [CrossRef]
2. Grant, I. Particle Image Velocimetry: A Review. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **1997**, *211*, 55–76. [CrossRef]
3. Scharnowski, S.; Kähler, C.J. Particle Image Velocimetry—Classical Operating Rules from Today's Perspective. *Opt. Lasers Eng.* **2020**, *135*, 106185. [CrossRef]
4. Scarano, F. Iterative Image Deformation Methods in PIV. *Meas. Sci. Technol.* **2001**, *13*, R1–R19. [CrossRef]
5. Astarita, T.; Cardone, G. Analysis of Interpolation Schemes for Image Deformation Methods in PIV. *Exp. Fluids* **2005**, *38*, 233–243. [CrossRef]
6. Huang, H.T.; Fiedler, H.E.; Wang, J.J. Limitation and Improvement of PIV. *Exp. Fluids* **1993**, *15*, 263–273. [CrossRef]
7. Wereley, S.T.; Meinhart, C.D. Second-Order Accurate Particle Image Velocimetry. *Exp. Fluids* **2001**, *31*, 258–268. [CrossRef]
8. Lecordier, B.; Demare, D.; Vervisch, L.M.J.; Réveillon, J.; Trinité, M. Estimation of the Accuracy of PIV Treatments for Turbulent Flow Studies by Direct Numerical Simulation of Multi-Phase Flow. *Meas. Sci. Technol.* **2001**, *12*, 1382. [CrossRef]

9. Hart, D.P. High-Speed PIV Analysis Using Compressed Image Correlation. *J. Fluids Eng.* **1998**, *120*, 463–470. [CrossRef]
10. Adrian, R.J. Double Exposure, Multiple-Field Particle Image Velocimetry for Turbulent Probability Density. *Opt. Lasers Eng.* **1988**, *9*, 211–228. [CrossRef]
11. Keane, R.D.; Adrian, R.J. Theory of Cross-Correlation Analysis of PIV Images. *Appl. Sci. Res.* **1992**, *49*, 191–215. [CrossRef]
12. Westerweel, J.; Draad, A.A.; van der Hoeven, J.G.T.; van Oord, J. Measurement of Fully-Developed Turbulent Pipe Flow with Digital Particle Image Velocimetry. *Exp. Fluids* **1996**, *20*, 165–177. [CrossRef]
13. Astarita, T. Analysis of Velocity Interpolation Schemes for Image Deformation Methods in PIV. *Exp. Fluids* **2008**, *45*, 257–266. [CrossRef]
14. Gui, L.; Wereley, S.T. A Correlation-Based Continuous Window-Shift Technique to Reduce the Peak-Locking Effect in Digital PIV Image Evaluation. *Exp. Fluids* **2002**, *32*, 506–517. [CrossRef]
15. Mollin, R.A. *Quadratics*, 1st ed.; CRC Press: Boca Raton, FL, USA, 1975.
16. Mendes, L.; Bernardino, A.; Ferreira, R.M.L. Piv-Image-Generator: An Image Generating Software Package for Planar PIV and Optical Flow Benchmarking. *SoftwareX* **2020**, *12*, 100537. [CrossRef]
17. Williamson, P.N.; Docherty, P.D.; Khanafer, A.; Steven, B.M. Analysis of Flow Through Extra-Anatomic Bypasses Between Supra-Aortic Branches Using Particle Image Velocimetry (PIV). *Clin. Med. Insights Cardiol.* **2024**, *18*, 11795468231221413. [CrossRef] [PubMed]
18. Stanislas, M.; Okamoto, K.; Kähler, C. Main Results of the First International PIV Challenge. *Meas. Sci. Technol.* **2003**, *14*, R63. [CrossRef]
19. Raffel, M.; Willert, C.E.; Scarano, F.; Kähler, C.J.; Wereley, S.T.; Kompenhans, J. *Particle Image Velocimetry: A Practical Guide*; Springer: Berlin/Heidelberg, Germany, 2018.
20. Keane, R.D.; Adrian, R.J. Optimization of Particle Image Velocimeters. I. Double Pulsed Systems. *Meas. Sci. Technol.* **1990**, *1*, 1202. [CrossRef]
21. Garbey, M.; Pacull, F. A Versatile Incompressible Navier–Stokes Solver for Blood Flow Application. *Int. J. Numer. Methods Fluids* **2007**, *54*, 473–496. [CrossRef]
22. Ryan, N.W.; Johnson, M.M. Transistion from Laminar to Turbulent Flow in Pipes. *AIChE J.* **1959**, *5*, 433–435. [CrossRef]