OpenOpticalFlow_PIV: An Open Source Program Integrating Optical Flow Method with Cross-Correlation Method for Particle Image Velocimetry

Tianshu Liu*, David M. Salazar

Department of Mechanical and Aerospace Engineering

Western Michigan University, Kalamazoo, MI 49008

(12/18/2019)

(Submitted to Journal of Open Research Software for consideration of publication)

*: The corresponding author, Department of Mechanical and Aerospace Engineering, G-217, Parkview Campus, Western Michigan University, Kalamazoo, MI 49008, USA

tianshu.liu@wmich.edu, 269-276-3426

Abstract

This paper describes OpenOpticalFlow_PIV, an open source Matlab program integrating the optical flow method with the cross-correlation method for extraction of high-resolution velocity fields from particle images. This hybrid method provides an additional tool to process PIV images, which combines the advantages of the optical flow method and cross-correlation method. The principles of the hybrid method are concisely described, including the cross-correlation method and optical flow method. The focus of this paper is the description of the main program, relevant subroutines and selection of the relevant parameters in computation. Examples are given to demonstrate the applications of the hybrid method.

Keywords: optical flow, cross-correlation, particle image velocimetry, Matlab

(1) Overview

Introduction

Particle image velocimetry (PIV) is a standard technique for global velocity measurements [1, 2]. From a pair of successive particle images, a displacement vector field is extracted by maximizing the spatial cross-correlation function in interrogation windows in the images. Various algorithms for PIV have been evaluated [3-5]. The displacement determined by the cross-correlation method in a window is a spatially-averaged displacement of a cluster of particles contained in a window. Thus, the cross-correlation method achieves the spatial resolution of a velocity field at one vector per a window.

On the other hands, the optical flow method has been applied to PIV images to obtain velocity fields theoretically at the spatial resolution of one vector per a pixel [6-14]. The optical flow method is a differential approach based on computations of the time derivative and the spatial gradient of the image intensity field, requiring that the displacements should be smaller than the characteristic length scale of tractable features in the image plane. In PIV images, the tractable features (e.g. particles) usually are 1-3 pixels, and thus the displacements should be smaller than about 3 pixels. Furthermore, particle images, which are actually the distributions of spatially spiky noises in the image intensity, are not ideal for temporal and spatial differentiations. When the displacements of particles are much larger than their sizes, the temporal and spatial differentiations of the image intensity cannot be correctly calculated. Thus, PIV images pose a challenge for application of the optical flow method.

To overcome this intrinsic problem for the optical flow method applied to PIV images with large displacements, a question is whether the optical flow method can be integrated with the cross-correlation method. The cross-correlation method performs well for PIV images where the displacements are suitably large, while the optical flow method is good to PIV images with sufficiently small displacements. An attempt has been made to develop a hybrid method [15]. It is found that the hybrid method combining the cross-correlation method with the optical flow method indeed gives consistently improved results in comparison with other methods in PIV.

The objective of this paper is to describe an open source Matlab program of the hybrid method for extraction of high-resolution velocity fields from PIV images. It provides an additional tool for researchers to process PIV images in their specific problems in relevant scientific and engineering fields. First, the principles of the cross-correlation method and optical flow method are briefly recapitulated. Then, the main program and the key subroutines are described, particularly on the selection of the relevant parameters in cross=correlation and optical flow computations. Next, to demonstrate the applications of this program, several examples based on simulated and experimental PIV images are presented.

Principles of the hybrid method

a. Main idea

The main procedures of the hybrid method are briefly described, as illustrated in Fig. 1. From a pair of successive PIV images (image 1 and image 2), the cross-correlation method is used for initial estimation of a coarse-grained displacement field since it is less sensitive to the displacement magnitude, random noise and illumination change in PIV images. Then, a shifted image 1 is generated by using a shifting scheme based on the coarse-grained displacement field, where the spatial resolution of the original image 1 is restored by using the interpolation. Further, from the shifted image 1 and the image 2, the residual displacement field is extracted by using the optical flow method since the residual displacement magnitude is very small. The corrected displacement field is reconstructed by superposing the coarse-grained displacement field and the residual displacement field. The procedures can be iterated for further improvement. Usually, a single iteration is sufficient for the correction.

b. Cross-correlation method

The application of the correlation method is the first step of the hybrid method. The cross-correlation method is physically intuitive [1, 2]. An image domain is decomposed into many small subdomains called interrogation windows. A domain-averaged displacement vector in a window is determined by maximizing the cross-correlation function. The cross-correlation function can be directly computed. Since the cross-correlation function is essentially a convolution integral, it can be evaluated via the fast Fourier transform (FFT) algorithm. There are many research and commercial algorithms for PIV [3-5]. In this work, in order to make a standalone Matlab program of the hybrid method, we adopt the modified functions from the open-source Matlab software PIVlab [16]. The structure and performance of PIVlab were described by Thielicke and Stamhuis [17]. For operational simplicity, the main parameters retained in PIVlab are the window sizes in multiple paths and the window overlapping percentages.

c. Optical flow method

The optical flow equation in the image plane for PIV images is written as [6]

$$\partial I / \partial t + \nabla \cdot \left(I \boldsymbol{u} \right) = f\left(x_1, x_2 \right), \tag{1}$$

where I is the normalized image intensity, $\boldsymbol{u} = \left(u_1, u_2 \right)$ is the optical flow, and $\nabla = \partial / \partial x_i$ ($i = 1,2$) is the gradient operator in the image plane $\left(x_1, x_2 \right)$. The boundary term $f = w_{ext} \left\langle C_{ext} \right\rangle_a \boldsymbol{n} \cdot \left(N\boldsymbol{U} \right) \Big|_{\Gamma_1}^{\Gamma_2}$ acts as a source/sink term representing the effect of particles accumulated within the cloud layer confined by the two control planes Γ_1 and Γ_2, where $\boldsymbol{U}$ is the particle velocity in the 3D object space, N is the number of particles per unit total volume, $\boldsymbol{n}$ is the unit vector normal to Γ_1 and Γ_2, C_{ext} is the extinction cross section, and w_{ext} is the corresponding correlation coefficient. The optical flow in Eq. (1) is defined as $\boldsymbol{u} = \left(u_1, u_2 \right) = \gamma \left\langle \boldsymbol{U}_{12} \right\rangle_N$, where $\left\langle \boldsymbol{U}_{12} \right\rangle_N$ is the light-path-averaged velocity of particles projected onto the image plane, and γ is a factor of projection. Essentially, Eq. (1) is a differential equation describing the projected motion of particles in the image plane. It is assumed that the net flux of particles through the control surfaces is zero, i.e., $\boldsymbol{n} \cdot \left(N\boldsymbol{U} \right) \Big|_{\Gamma_1}^{\Gamma_2} = 0$.

To determine the optical flow as an inverse problem from Eq. (1), a variational formulation with a smoothness constraint is typically used [6, 18, 19]. Given I and f, we define a functional

$$J(\boldsymbol{u}) = \int_D \left(\partial I / \partial t + \nabla \cdot (I\,\boldsymbol{u}) - f\right)^2 dx_1 dx_2 + \alpha \int_D \left(|\nabla u_1|^2 + |\nabla u_2|^2\right) dx_1 dx_2 \, , \tag{2}$$

where the first integral term is the equation functional, the second integral term is the first-order Tikhonov regularization functional, D is an image domain of interest, and α is the Lagrange multiplier. Minimization of the functional $J(\boldsymbol{u})$ leads to the Euler-Lagrange equation

$$I \nabla \left[\partial I / \partial t + \nabla \cdot (I\boldsymbol{u}) - f\right] + \alpha \nabla^2 \boldsymbol{u} = 0 \, , \tag{3}$$

where $\nabla^2 = \partial^2 / \partial x_i \partial x_i$ is the Laplace operator, and α is the Lagrange multiplier that controls the smoothness of the field. When a pair of sequential images is given and f is neglected in the first-order approximation, the standard finite difference method is used to solve this partial differential equation with the Neumann condition $\partial \boldsymbol{u} / \partial n = 0$ on the image domain boundary ∂D for the optical flow $\boldsymbol{u}$. In theory, the optical flow method is able to extract a velocity field at the spatial resolution of one vector per a pixel. However, the actual spatial resolution is somewhat affected by the Lagrange multiplier that acts as an effective diffusion coefficient, i.e., the larger Lagrange multiplier tends to smooth out finer features. The optical flow algorithm used in this work is the same as the open source gramgam "OpenOpticalFlow.v1" [14]. The program has the Horn-Schunck estimator for an initial solution [18] and the Liu-Shen estimator for a refined solution [6, 19]. Optical flow computation sensitivity to the Lagrange multiplier is not large.

d. Parameters related to error

The main parameters related to the errors in PIV are the particle displacement magnitude $\left\|\Delta \boldsymbol{x}_p\right\|$, the particle image diameter d_p, the particle velocity gradient $\left\|\nabla \boldsymbol{u_p}\right\|$, and the particle image density N_p. The error is proportional to the particle displacement magnitude $\left\|\Delta \boldsymbol{x}_p\right\|$. The total error of the optical flow method applied to PIV images is given by

$$\varepsilon = \left\|\Delta \boldsymbol{x}_p\right\| \sqrt{\frac{c_1 d_p^2}{\left\|\boldsymbol{u}_p\right\|^2} + c_2 \left\|\nabla \boldsymbol{u}_p\right\|^2 + \frac{c_3}{d_p^2} + \frac{c_4}{N_p^m} \frac{\left\|\Delta^m \boldsymbol{x}_p\right\|^2}{\left\|\Delta \boldsymbol{x}_p\right\|^2}}\,, \tag{4}$$

where c_1, c_2, c_3 and c_4 are empirical coefficients [13].

The effect of the particle image diameter d_p in Eq. (4) is represented by the terms of $\sim d_p^2$ and $\sim d_p^{-2}$ that have the opposite trends as d_p increases, which indicates that there could be the optimum .diameter. Generally, the error decrease with increasing the particle image density N_p. But when N_p becomes very large, the PIV images become more uniform, and thus extraction of the optical flow is not accurate due to the very small intensity gradient. Therefore, based on this observation, there would be an optimal value of N_p. For the convenience of application, the particle displacement $\left\|\Delta \boldsymbol{x}_p\right\|$ is replaced by the maximum particle displacement $max\left(\left|\Delta \boldsymbol{x}_p\right|\right)$, and the particle velocity gradient magnitude $\left\|\nabla \boldsymbol{u_p}\right\|$ is replaced by the averaged magnitude of the particle velocity gradient $\left|\nabla \boldsymbol{u_p}\right|$.

e. Shifting scheme

The shifting scheme in the image plane is a necessary element of the hybrid method, which is used to generate the shifted image 1 for optical flow computation. The shifting scheme has two procedures. The first procedure is pixel-to-pixel mapping of the intensity of a given image onto the shifted image denoted by I_0 based on a given velocity field in pixels per unit time.

To include the contribution of the subpixel motion, the second procedure is described below. Since the coarse-grained velocity field obtained using the correlation method has a much lower spatial resolution, data interpolation is used to generate a high-resolution velocity field with the same size as the original images. For the subpixel velocity components δu_x and δu_y in the image plane, they follow the equation $\partial_t I + \partial_x \left(I\, \delta u_x \right) + \partial_y \left(I\, \delta u_y \right) = 0$ [20]. Integration of this equation from 0 to 1 (one unit time) gives the intensity variation associated with the subpixel motion $\delta I = -\Delta_x \left(I\, \delta u_x \right) - \Delta_y \left(I\, \delta u_y \right)$, where Δ_x are Δ_y the differences over one pixel in the *x* and *y* coordinates in the image plane, and the spatial differences are $\Delta x = \Delta y = 1$ pixel. Therefore, the corrected mapping $I\left(i, j \right) \rightarrow I_0 + \delta I$ is used in the shifting scheme, where I_0 denotes the pixel-to-pixel shifted image obtained using the first procedure.

f. Pre-processing scheme

A pre-processing scheme is made to correct illumination intensity change. The laser illumination intensity may change between two successive PIV images acquired, which will affect optical flow computation. The illumination intensity change is contributed by an overall shift and a local change. The scheme has two steps. In the first step, the overall change in the whole images is corrected by shifting the averaged intensity difference between the two images.

The second step is to correct the local illumination intensity change. By applying a Gaussian filter with a suitable standard deviation to the two images, two sufficiently filtered (smoothed) images are obtained. The difference between the two filtered images represents the local illumination intensity change, and it is used to compensate the image intensity change by adding the difference to the original image to be corrected. The selection of the standard deviation σ of the Gaussian filter is a key, depending on the characteristic length scale of the local change pattern. When σ is too small, the optical flow could not be extracted correctly. On the other hand, when σ is too large, the local change could not be sufficiently corrected. Therefore, trial-and-error tests should be made to find a suitable value of σ for a particular application.

Implementation and architecture

a. General description

The program of the hybrid method is written in Matlab, and the files inclusing the main program and subroutines are contained in a folder named "OpenOpticalFlow_PIV.v1". The main program is "Hybrid_Flow_Diagnostics_Run.m". As shown in Fig. 1, the main program has the following steps:

(1) A pair of images (Image 1 and Image) is loaded;

(2) The relevant parameters are set;

(3) The images are pre-processed;

(4) A coarse-grained displacement field is extracted from Image 1 and Image 2 by using a cross-correlation method;

(5) A shifted image 1 is generated from Image 1 using the shifting scheme based on the coarse-grained displacement field that is interpreted at every pixel;

(6) The residual displacement field is obtained from the shifted image 1 and Image 2 by using the optical flow method;

(7) The corrected displacement field is obtained by superposing the interpreted coarse – grained displacement field and the residual displacement field;

(8) The results including the velocity vectors, velocity magnitude, streamlines, vorticity and second invariant are plotted.

The coarse-grained displacement field for the initial approximation is obtained by using the subroutine "pivAnalyzeImagePair" that is adopted from the open-source Matlab software PIVlab [16, 17]. In optical flow computation, the subroutine for solving Eq. (3) is "liu_shen_estimator.m", while the solution of the Horn-Schunck optical flow equation ("horn_schunck_estimator.m") is used as an initial approximation for Eq. (3) for faster convergence. The combination of "liu_shen_estimator.m" and "horn_schunck_estimator.m" constitutes the key elements of the optical flow program.

The processing can be made in a selected rectangular domain of interest for regional flow diagnostics when the index "index_region" is set at 1. Otherwise, the processing is conducted in the whole image domain when the index "index_region" is set at 0. The input files and relevant parameters for computation are listed in Table 1. The image files "Im1" and "Im2" could be in the conventional image format such as "tif", "bmp", or "jpeg". The input parameters are discussed in the following subsections. The optical flow $\boldsymbol{u} = (u_1, u_2)$ is calculated by solving Eq. (3). Table 2 lists the outputs. The output data are "ux0" and uy0" that are the coarse-grained velocity components in the image plane (x, y), which are extracted by using the cross-correlation method. The refined velocity components "ux_corr" and "uy_corr" are obtained by using the optical flow method. The unit of the velocity given by the program is pixels/unit-time, where the unit time is the time interval between two input images. The velocity can be converted to the physical unit of m/s after the relation between the image plane and the 3D object space is established through camera calibration/orientation. The vorticity, strain rate and second invariant can be calculated based on a high-resolution velocity field. Furthermore, the main program can be adapted as a subroutine in a loop for processing of a sequence of image pairs such that the statistical quantities of the flow can be obtained.

b. The Lagrange multipliers

In optical flow computation, the Horn-Schunck estimator (“horn_schunck_estimator.m”) is used for an initial solution, and Liu-Shen estimator (“liu_shen_estimator.m”) is used for a refined solution of Eq. (1). In the main program, the Lagrange multipliers “lambda_1” and “lambda_2” are selected for the Horn-Schunck and Liu-Shen estimators, respectively. For example, “lambda_1” = 20 and “lambda_2” = 2000 for typical computations. Since no rigorous theory exists for determining the Lagrange multiplier, simulations based on a synthetic velocity field for a specific measurement are conducted to determine the Lagrange multiplier by using an optimization scheme [8]. The smallest Lagrange multiplier that still leads to a well-posed solution could be selected as a starting value in a trial-and-error process. Nevertheless, within a considerable range of the Lagrange multipliers, the solution is not significantly sensitive to its selection. Users can do some tests by adjusting the Lagrange multipliers for their applications.

c. Filtering

Pre-processing of images is sometimes required to remove the random noise by using a Gaussian filter, where the standard deviation (std) of a Gaussian filter is selected, depending on the noise level in a specific application (for example the std of a Gaussian filter is 4-6 pixels for images of 480×520 pixels). In the main program, the mask size of a Gaussian filter is given by the parameter called “size_filter”, and the std of the Gaussian filter is 0.6 of the mask size. In certain applications, the background noise in PIV images should be cleaned up by setting the cutting threshold value.

d. Correction for illumination change

In optical flow computation, it is assumed that the illumination light intensity is constant. However, the illumination intensity field could change in a time interval between two successively acquired PIV images. In this case, correction for this illumination intensity change is required before optical flow computation. This correction is done by calling the subroutine “correction_illumination”. As described in the previous section (f), the first step is the overall illumination change in the whole image is corrected by shifting an averaged intensity value in a selected domain. The parameter “window_shifting,size” defines the rectangular domain for calculating the averaged intensity value. The second step is the use of a simple scheme for correcting local illumination intensity changes based on the application of a Gaussian filter [11]. The selection of the std (or size) of a Gaussian filter is important to determine the local averaged intensity value for correction. This size of the Gaussian filter is given by the parameter called “size_average” in the main program. The selection of the filter size is a trial-and-error process depending on the pattern of illumination change in a specific measurement, and simulations on a synthetic velocity field are helpful to determine the suitable size.

e. Coarse-grained displacement field

The coarse-grained displacement field for the initial approximation is obtained by using the cross-correlation subroutine “pivAnalyzeImagePair”. The window sizes in the multiple-path process are given by the parameter “pivPar.iaSizeX”. For example, “pivPar.iaSizeX = [64 32 16]” defines the three-path process from 64×64 to 32×32 to 16×16 pixel^2 windows. The overlapping between the windows is defined by the parameter “pivPar.iaStepX”. For example, the 50% overlapping is defined by “pivPar.iaStepX = [32 16 8]” which corresponds to “pivPar.iaSizeX = [64 32 16]”. The parameter for the method of cakculating the cross-correlation is “pivPar.ccMethod = 'fft'”, indicating the FFT is used.

In principle, the displacement field can be successively improved by iterations to achieve a better accuracy. The iteration number for successive corrections is given by “no_iteration”. Since one correction is usually sufficient, “no_iteration” is set at 1. When “no_iteration” is set at 0, the correction is disabled. Due to data interpretation, the data near the left and top edges have large errors. These bad data are replaced by setting the data using the good data in the nearest interior region. The edge width is defined by the parameter “edge_width”.

Examples

a. Oseen vortex pair in uniform flow

An Oseen vortex pair in uniform flow has been used as a canonical case to evaluate the accuracy of the optical flow method [6, 13]. Here this flow is used as an example for application of the hybrid method. A pair of the particle image with the size of 500×500 pixels and the 8-bit dynamic range is generated for simulations, where 10000 particles with a Gaussian intensity distribution with the standard deviation of 2 pixels are uniformly distributed. Two Oseen vortices are placed at $(m/3, n/2)$ and $(2m/3, n/2)$ in an image, respectively, where m (500) and n (500) are the numbers of rows and columns of the image. The vortex strengths are $\Gamma = \pm 7000$ $(\text{pixel})^2$/s and the vortex core radius is $r_0 = 15$ pixels. The uniform flow velocity is

10 pixels/s. The second image is generated by using the shifting scheme based on the given velocity field after a time step. Figure 2 shows the generated image 1 and image 2, where the mean characteristic image diameter of particles is 4 pixels and the particle image density is 0.04 1/pixel^2.

The hybrid method is applied to Image 1 and Image 2. The cross-correlation method is applied with the three-path process from 32×32 to 16×16 to 8×8 and the 50% overlapping. In optical flow computation, the Lagrange multipliers in the Horn-Schunck estimator and the Liu-Shen estimator are set at 20 and 2000, respectively. Figure 3 shows the fields of the velocity vector, velocity magnitude and vorticity extracted from the particle images, where the streamlines are also shown and the velocity magnitude and vorticity field normalized by their maximum value.

The accuracy of the hybrid method has been studied through simulations of the Oseen vortex pair in in a uniform flow comparison with the optical flow method and cross-correlation methods [20]. As indicated before, the error depends on the four parameters: particle displacement, particle velocity gradient, particle image density, and particle image diameter. It is found that the hybrid method consistently performs better than the typical cross-correlation methods although the optical flow method is more accurate for PIV images with sufficiently small displacements (typically less than 3 pixels). To large extent, the hybrid method indeed combines the advantages of the cross-correlation method and the optical flow method for PIV measurements.

b. Vortex-induced separation

PIV images were obtained in a circular air jet normally impinging on a wall from a contoured circular nozzle in experiments [13]. Figure 4 shows a pair of PIV images focusing on the wall-jet region where strong interactions between vortices and boundary layer occur. Figure 5 shows the fields of the velocity vectors, velocity magnitude and vorticity obtained by using the hybrid method. It is revealed that large vortices generated in the shear layer of the wall jet due to the Kelvin-Helmholtz instability induce secondary vortices (secondary separations) in boundary layer near the wall. Compared to the cross-correlation method in PIV, the hybrid method yields the velocity field with much higher spatial resolution (one vector per a pixel), revealing more details of the near-wall flow structures in the thin wall jet (the thickness is about 3 mm in the physical scale, and 100 pixels in the image plane).

c. Twin jet

PIV images were obtained in twin jet excited acoustically where the ratio between the space between the two circular jets and exit diameters was 3 and the Reynolds number based on the diameter was 1800 [21]. Figure 6 shows a pair of particle images of acoustically excited twin jet. Figure 7 shows the fields of velocity vectors, velocity magnitude, and vorticity of acoustically excited twin jet extracted from the particle images.

(2) Availability

Operating system

Based on Matlab (R2007a, or later versions): Windows

Programming language

Matlab (R2007a, or later versions)

Additional system requirements

None

Dependencies

Several functions in Matlab image processing toolbox are required.

List of contributors

Tianshu Liu, Department of Mechanical and Aerospace Engineering, Western Michigan University

David M. Salazar, Department of Mechanical and Aerospace Engineering, Western Michigan University

Software location

Name: GitHub

Identifier: https://github.com/Tianshu-Liu/OpenOpticalFlow_PIV

License: MIT license

Data published: December 16, 2019

(3) Reuse Potential

Global velocity diagnostics is of fundamental importance in the study of fluid mechanics in order to understand the physics of complex flows. Integration of the optical flow method with the cross-correlation method allows extraction of high-resolution velocity fields from PIV images in various measurements in different scientific and engineering applications. The hybrid optical-flow-correlation method combines the advantages of the optical flow method and correlation method for PIV images with large displacements [20, 21]. Computation using this method can be conducted in a PC with Matlab to extract velocity fields at a spatial resolution of one vector per a pixel. This open source program, OpenOpticalFlow_PIV, allows users to adapt the hybrid method for their specific problems. Furthermore, OpenOpticalFlow_PIV can be integrated with Matlab for convenient image processing, data presentations, and data input/output. Besides GitHub, the programs can be directly downloaded from author's website https://wmich.edu/mechanical-aerospace/directory/liu.

References:

[1] Raffel, M., Willert, C., Wereley, S. & Kompenhans, J. 2007 *Particle image velocimetry* Springer, Berlin, Germany.

[2] Adrian, R. J., & Westerweed, J. 2011 *Particle image velocimetry* Cambridge University Press, Cambridge, UK.

[3] Stanislas, M., Okamoto, K. & Kähler, C. 2003 Main results of the first international PIV challenge *Meas Sci Technol* 14:R63-R89.

[4] Stanislas, M., Okamoto, K., Kähler, C. & Westerweel, J. 2005 Main results of the second international PIV challenge *Exp Fluids* 39 170-191.

[5] Stanislas, M., Okamoto, K., Kähler, C. & Westerweel, J. Scarano F 2008 Main results of the third international PIV challenge *Exp Fluids* 45 27-71.

[6] Liu, T, & Shen, L. 2008 Fluid flow and optical flow *J Fluid Mech* 614:253-291.

[7] Heitz, D., Memin, E. & Schnorr, C. 2010 Variational fluid flow measurements from image sequences: synopsis and perspectives *Exp Fluids* 48 369-393.

[8] Quenot, G. M., Pakleza, J. & Kowalewski, T. A. 1998 Particle image velocimetry with optical flow *Exp Fluids* 25 177-189.

[9] Ruhnau, P., Kohlberger, T., Schnorr. C. & Nobach, H. 2005 Variational optical flow estimation for particle image velocimetry *Exp Fluids* 38 21-32.

[10] Yuan, J., Schnorr, C. & Memin, E. 2007 Discrete orthogonal decomposition and variational fluid flow estimation *Journal of Mathematical Imaging and Vision* 28 67-80.

[11] Corpetti, T., Memin, E. & Perez, P. 2002 Dense estimation of fluid flows *IEEE Trans on Pattern Analysis and Machine Intelligence* 24 365-380.

[12] Corpetti, T., Heitz, D., Arroyo, G. & Memin, E. 2006 Fluid experimental flow estimation based on an optical flow scheme *Exp Fluids* 40 80-97.

[13] Liu, T, Merat, A., Makhmalbaf, MHM., Fajardo, C., & Merati, P. 2915 Comparison between optical flow and cross-correlation methods for extraction of velocity fields from particle images *Exp Fluids* 56 166-189.

[14] Liu, T. 2017 OpenOpticalFlow: An open source program for extraction of velocity fields from flow visualization images *Journal of Open Research Software* 5 29.

[15] Yang, Z., & Johnson, M. 2017 Hybrid particle image velocimetry with the combination of cross-correlation and optical flow method *Journal of Visualization* 20(3) 625-638.

[16] Vejražka, J. 2019 Private communication.

[17] Thielicke, W. & Stamhuis, E. J. 2014 PIVlab – Towards user-friendly, affordable and accurate digital particle image velocimetry in MATLAB *Journal of Open Research Software* 2(1) e30.

[18] Horn, B. K. & Schunck, B. G. 1981 Determining optical flow *Artificial Intelligence* 17 185-204.

[19] Wang, B., Cai, Z., Shen, .& Liu, T. 2015 An analysis of physics-based optical flow method *J Comp Appl Math* 276 62-80.

[20] Liu, T., Salazar, D. M., Fagehi, H., Ghazwani, H., Montefort, J., & Merati, P., 2019 Hybrid optical-flow-cross-correlation method for particle image velocimetry, Journal of Fluids Engineering (in press).

[21] Salazar, D. Hybrid optical-flow-cross-correlation method for particle image velocimetry, MS Thesis, Department of Mechanical and Aerospace Engineering, Western Michigan University, December 2019.

Table 1. Inputs

Input files and parameters	**Notation**	**Unit**	**Note**
image pair	Im1, Im2	-	tif, bmp,jpeg etc.
Lagrange multipliers	lambda_1 for the Horn-Schunck estimator lambda_2 for the Liu-Shen estimator	-	regularization parameter in variational solution
Gaussian filter size	size_filter	pixel	removing random image noise
Gaussian filter size	size_average	pixel	correction for local illumination intensity change
scale factor for downsampling of original images	scale_im	-	reduction of initial image size in coarse-to-fine scheme
number of iterations	no_iteration	-	iteration in coarse-to-fine scheme (≥ 1)
left and upper edge width	edge_width	pixel	clean left and upper edges where abnormal data may occur
indicator for regional diagnostics (0 or 1)	index_region	-	"0" for whole image; "1" for selected region
sizes of interrogation area	pivPar.iaSizeX	pixel	defining multi-path of windows
grid spacing	pivPar.iaStepX	pixel	defining overlapping
method for calculating cross-correlation	pivPar.ccMethod		such as 'fft'

Table 2. Outputs

Output files	**Notation**	**Unit**	**Note**
coarse-grained velocity	ux0, uy0	pixels/unit-time	based on downsampled images
refined velocity	ux_corr, uy_corr	pixels/unit-time	refined result with full spatial resolution in coarse-to-fine process

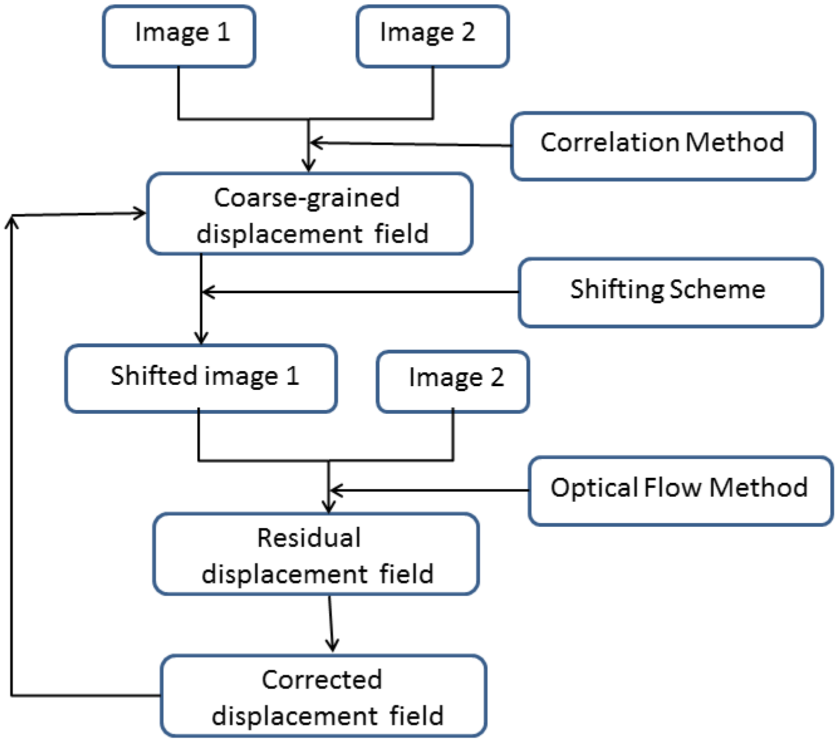
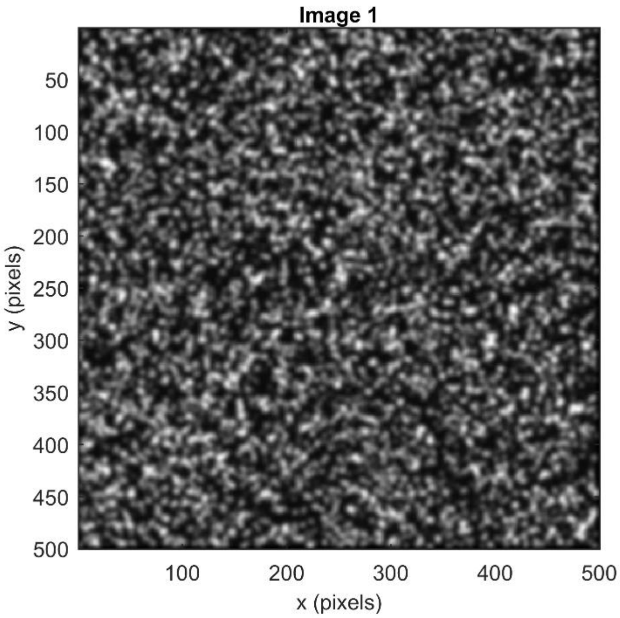
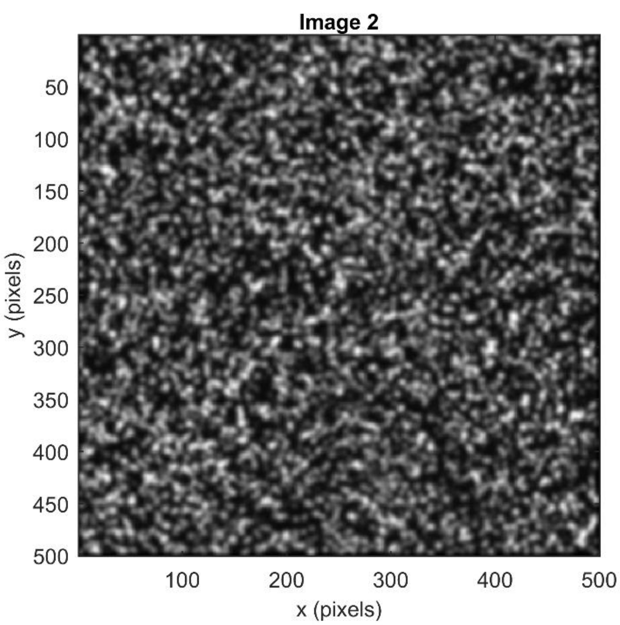


Figure 1. Block diagram for the hybrid method.

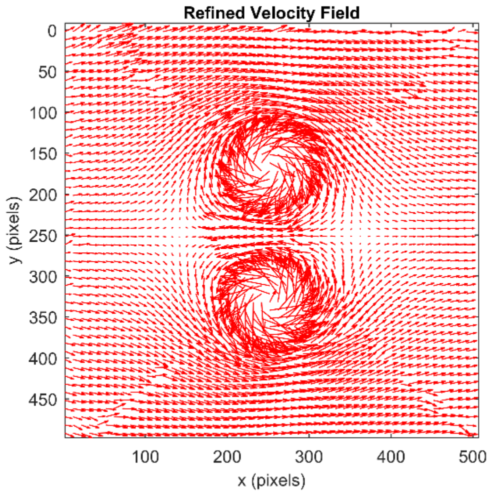


(a)

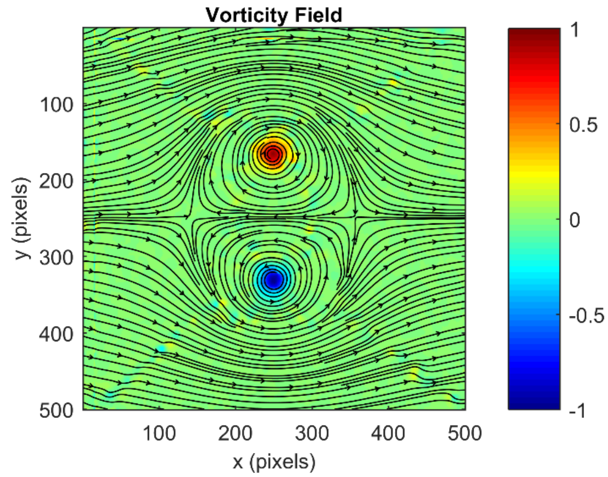


(b)

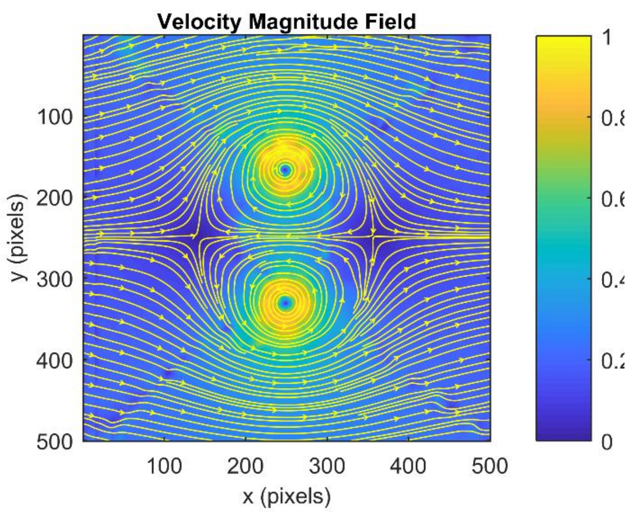
Figure 2. A pair of particle images of an Oseen vortex pair in a uniform flow, (a) Image #1, and (b) Image #2.



(a)

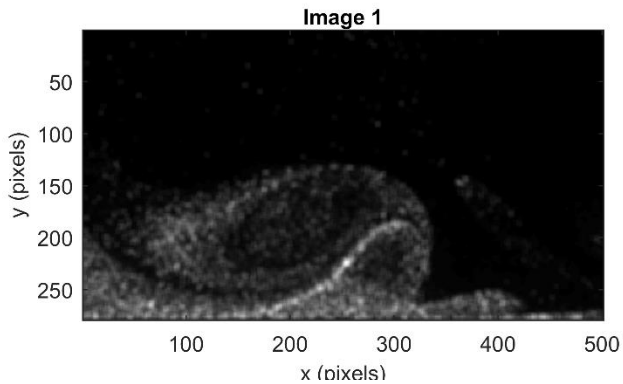


(c)

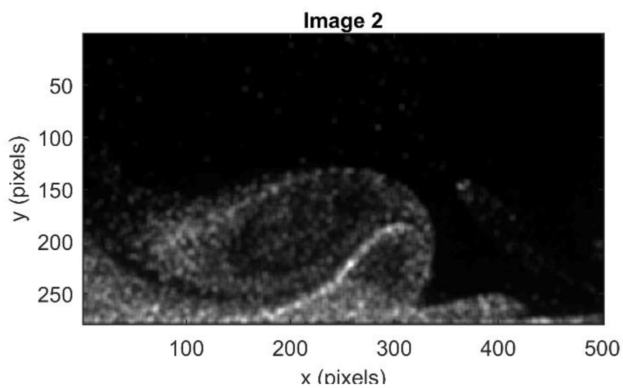


(b)

Figure 3. Flow fields of an Oseen vortex pair in a uniform flow extracted from the particle images, (a) velocity vectors, (b) velocity magnitude, and (c) vorticity.

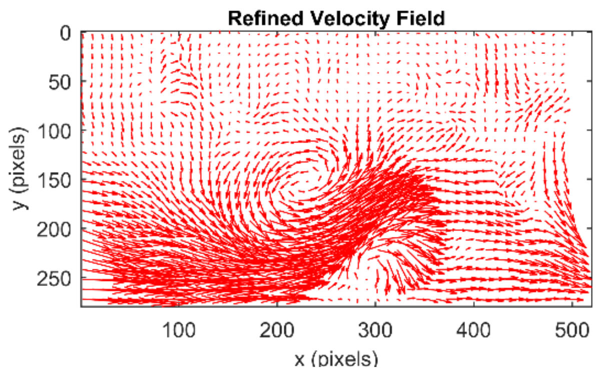


(a)

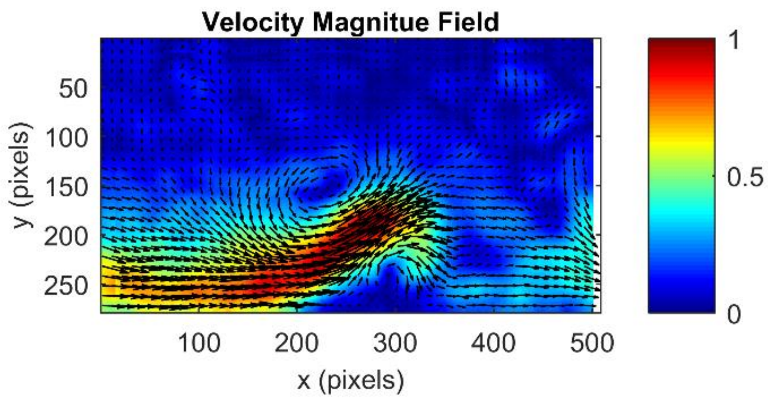


(b)

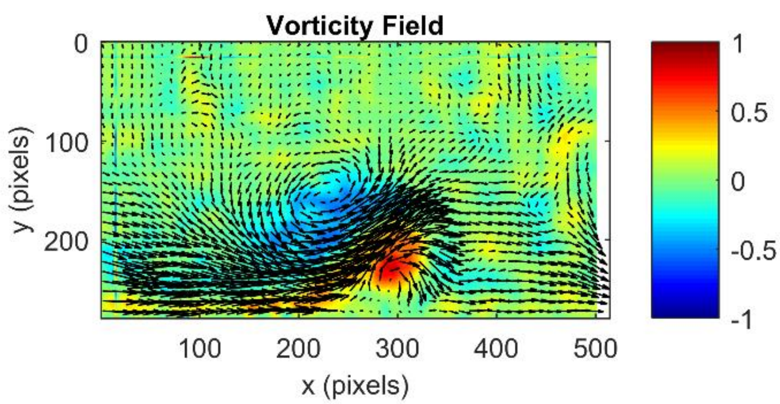
Figure 4. A pair of particle images of vortex-induced separation in a wall-jet region of a normally impinging jet, (a) Image #1, and (b) Image #2.



(a)

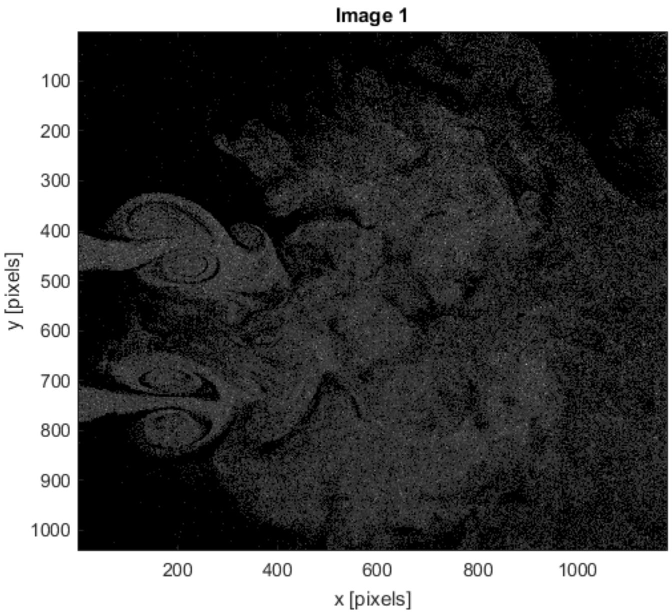


(b)

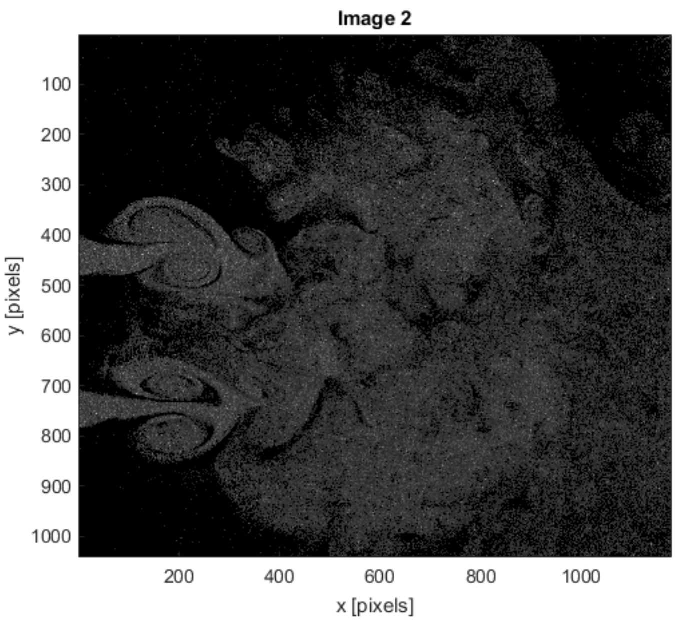


(c)

Figure 5. Flow fields of vortex-induced separation in a wall-jet region of a normally impinging jet extracted from the particle images, (a) velocity vectors, (b) velocity magnitude, and (c) vorticity.



(a)



(b)

Figure 6. A pair of particle images of acoustically excited twin jet, (a) Image #1, and (b) Image #2.

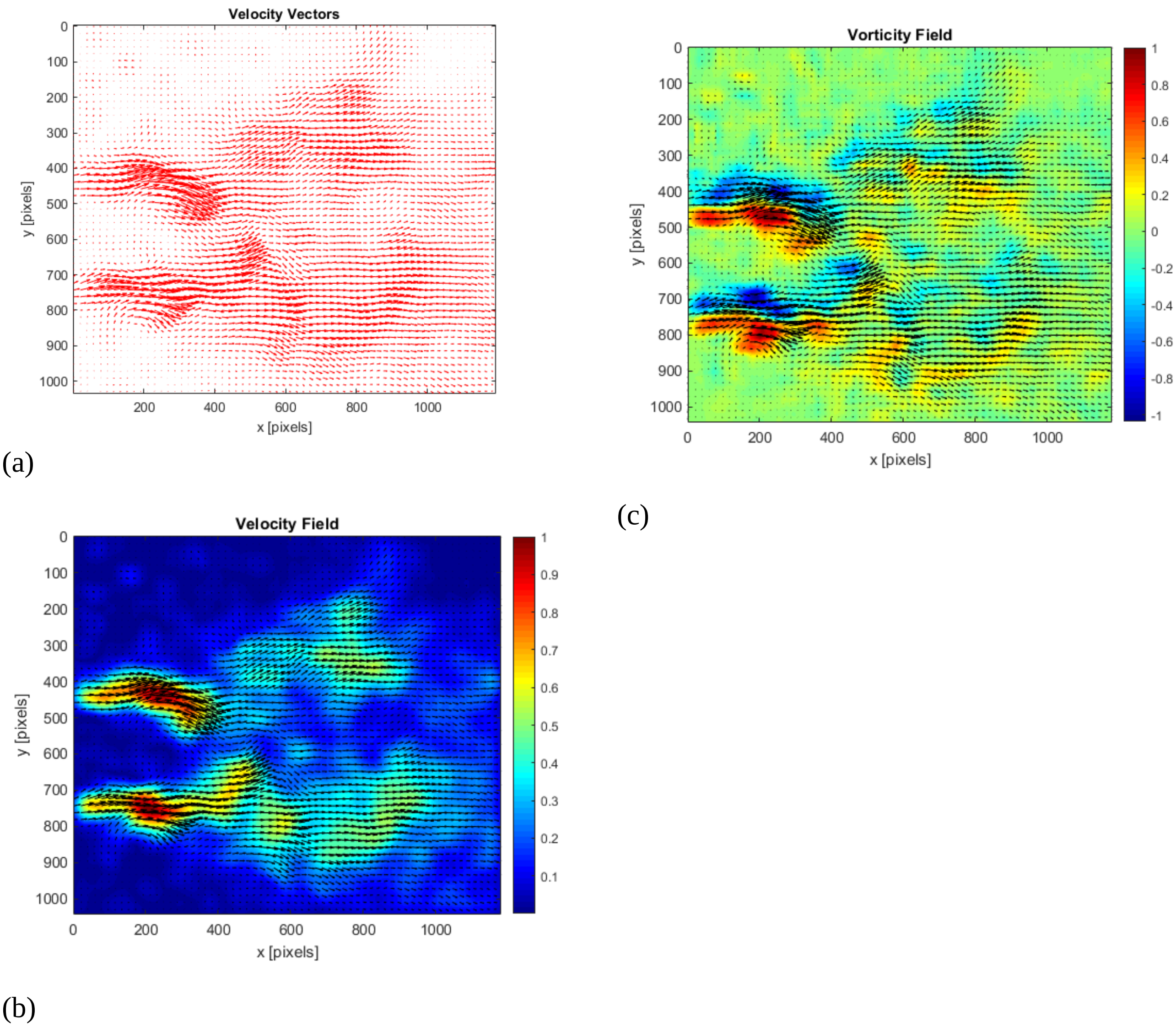


Figure 7. Flow fields of acoustically excited twin jet extracted from the particle images, (a) velocity vectors, (b) velocity magnitude, and (c) vorticity.