Journal of Flow Visualization and Image Processing 27(4):359–376 (2020)

DEVELOPING PARTICLE IMAGE VELOCIMETRY SOFTWARE BASED ON A DEEP NEURAL NETWORK

Wojciech Majewski,[1,] Runjie Wei,[1] & Vivek Kumar[2]*

[1]*Microvec Pte Ltd., Singapore*

[2]*Indian Institute of Technology, Kanpur, India*

*Address all correspondence to: Wojciech Majewski, Microvec Pte Ltd., Singapore, E-mail: wmajewski@piv.com.sg

As an experimental technique for fluid mechanics, particle image velocimetry (PIV) can extract global and quantitative velocity field from images. With the development of artificial intelligence, designing PIV method based on deep learning is quite promising and worth exploring. First, in this paper, the authors introduce the optical flow neural network based on one proposed in the computer vision community. Second, a data set including particle images and the ground truth fluid motion is generated to train the parameters of the networks. This leads to a deep neural network for PIV which can provide estimation of dense motion (down to maximum one vector for one pixel) with the high degree of efficiency. The featuring of particle image extracted by the neural network is also investigated in this paper. It is found that feature matching improves the accuracy of estimation. The proposed network model is firstly evaluated by a synthetic image sequence of turbulent flow. An experiment measuring the flow over an aerofoil is used to validate the practicability. The experimental results indicate that compared with the traditional cross correlation method, the proposed deep neural network has advantages in accuracy, spatial resolution, and efficiency.

KEY WORDS: *particle image velocimetry, estimation of fluid motion, deep neural networks*

1. INTRODUCTION

Particle image velocimetry (PIV) is a nonintrusive technique for quantitative measurement of the velocity fields (Adrian et al., 2011; Raffel et al., 2018). PIV can analyze the velocity field information of fluid motion from the images and visualize it, helping researchers to understand more complex flow phenomena. There are many PIV methods like 2D planar PIV, which can be used for measuring two-dimensional flows in a plane or tomographic PIV, which can be used for volumetric flow measurements.

Since 1984, PIV has become one of the standard techniques for flow measurements and has been widely adopted in fluid mechanics research. The principle has not changed drastically in the last 35 years. Seeding particles are inserted into the

1065-3090/20/$35.00

flow, and the inherent assumptions are, that they follow the fluid motion exactly, that they are distributed homogeneously and their displacement within interrogation region is uniform. Double-pulse lasers, capable of sending two pulses within a very short time, are used to generate light sheet with special optics. This laser light illuminates a plane, making the seeding particles to follow the visible flow. The light scatters off the particles, and an image is taken by a camera, capturing an instantaneous snapshot of the flow showing the particles clearly. A second image is taken shortly afterwards, illuminated by the second pulse of the PIV laser, and the displacement of the particles between these two images can be seen. By applying a cross correlation algorithm (Raffel et al., 2018) on the grid of interrogation regions as low as 16 × 16 pixels, the velocity vectors can be calculated showing the direction and speed of the flow for each interrogation region. When the analysis for the entire image has been completed, the resulting velocity field, vorticity field, and many other flow characteristics can be seen and analyzed.

Currently multigrid iterative calculation based on image deformation is the most common method in use (WIDIM for short) (Scarano, 2005), the algorithm reduces the interrogation region size step by step. Starting from 64 × 64 pixels down to 16 × 16 pixels gradually the number of vectors is getting higher, and the calculated velocity field resolution results in a higher precision field. Most traditional commercial and free PIV software is based on this model. The FFT correlation analysis calculates the statistical average displacement within the window, is limited by the size of the interrogation region. There were many attempts to bring the size of the interrogation region to a lower size like 8 × 16 or even 8 × 8, but the calculated results have not been satisfactory as shown by Marusic (2019) in his keynote paper presented at the 2019 ISPIV in Munich. The resolution of the vector field obtained in such a way is not sufficient to see and understand the turbulent flows in the boundary layers.

Another common algorithm used in PIV is an optical flow method (Horn et al., 1981). It overcomes the limitation of interrogation region by its ability to extract the velocity field at the maximum single pixel level. The optical flow method originated in the field of computer vision. It relies on the optimization of an objective function, with the assumption of illumination invariance and the assumption of velocity field smoothing. The objective function is constructed, and the corresponding velocity field is obtained by minimizing the variational function of the objective function. The variational solution is solved to obtain the corresponding velocity field.

Similar to the WIDIM correlation analysis, the more complete optical flow method is usually linearized around current estimates and embedded into a multiresolution pyramidal image structure obtained from successive low-pass filtering and subsampling of the image sequence (Ruhnau et al., 2005; Heitz et al., 2010), which can satisfy large particle displacement estimation. The estimation process is then incrementally conducted from "coarse to fine" along the multiresolution structure.

In recent years, the optical flow method has received increasing attention in the field of PIV. The researchers established the relationship between optical flow and physical properties of fluids, thereby improving the accuracy of the estimation of fluid motion by the optical flow method (Corpetti et al., 2006; Liu and Shen, 2008; Chen et al., 2015). However, due to the need for variational optimization, the computational efficiency of the optical flow method is low. In addition, the hyper-parameters that are difficult to adjust, are limiting further the expansion and application of the optical flow method.

To overcome the limitations of the above correlation analysis and optical flow method, this paper shows a PIV algorithm based on deep learning to achieve maximum single-pixel, accurate and fast velocity field estimation. Deep learning has had great success in the computer field (LeCun et al., 2015; Krizhevsky et al., 2012). Prior to this, some research work combined PIV with deep neural networks (Rabault et al., 2017; Lee et al., 2017). These papers proposed the work of using a multilevel level regression, deep convolutional neural networks. At each level, the networks are trained to predict a vector from input image pairs to form a similar correlation analysis, that is, using a neural network to estimate a velocity vector in a particle image window. The low-level network is applied to estimation of large displacement, and the high-level networks are used to improve the spacial accuracy. The latest research results (Cai et al., 2019a,b) based on optical flow neural network, improve the design further by extracting dense velocity field, and have high computational efficiency when satisfying certain precision.

The connection between fluid flow and optical flow is explored in typical flow visualizations to provide a rational foundation for application of the optical flow method to image-based fluid velocity measurements. The projected-motion equations are derived, and the physics-based optical flow equation is given. In general, the optical flow is proportional to the path-averaged velocity of fluid or particles weighted with a relevant field quantity. The variational formulation and the corresponding Euler–Lagrange equation are given for optical flow computation. An error analysis for optical flow computation is provided, which is quantitatively examined by simulations on synthetic grid images. Direct comparisons between the optical flow method and the correlation-based method are made in simulations on synthetic particle images and experiments in a strongly excited turbulent jet.

1.1 Introduction to Deep Neural Network

This paper further enhances the above literature. The starting point is a convolution neural network (CNN) proposed by researchers in the field of computer vision, because of its highest level of accuracy in estimation of the rigid body motion (Hui et al., 2018). In order to adapt it to the fluid motion measurement, the novel network structure is implemented, by partially modifying and adjusting the training parameters. At the same time, the artificially generated PIV data set is used for network train-

ing to obtain native deep neural network for PIV. The trained network model is used to test the experimental data and the real experimental data evaluation. This paper focuses on calculating the image velocity field. The test results show that the proposed PIV algorithm based on deep neural network has obvious advantages and prospects in accuracy, resolution, and computational efficiency.

As shown in Fig. 1, the original network structure consists of two subsections: a compression encoder and an extension decoder (Cai et al., 2019b). The encoder is used for feature extraction of input images, while the decoder implements optical flow field estimation from "coarse to fine" for these features.

The pyramid feature extraction encoder consists of two network threads that compress two input images to form two pyramids of multiscale high-dimensional features. This function is implemented by a set of convolutional layers, with image pair input $\left\{F_{1,2}^{k}\right\}$. The subscript is the image number, and k is the feature at different scales (for example, F^0 is the highest layer with full resolution, and F^1 is the second layer with 1/2 resolution). There are 6 pyramid feature levels. Therefore, the encoder contains 6 layers of down sampling convolution operations.

The optical flow estimation decoder estimates the velocity field from coarse to fine by using a multilevel deconvolution layer. At each pyramid level (for example, the second layer in Fig. 1), by using the same resolution as the encoder. The feature (i.e., F^2) and the estimated velocity field from the previous layer (i.e., the third layer of $\dot{x}$). For example, in the second layer of the decoder, this function can be simply expressed as

$$\dot{x}^2 = E\left(F_1^2, F_2^2, \dot{x}^3\right). \tag{1}$$

Among them, the function E of the decoder represents a series of operations at each pyramid level, which can be summarized as follows:

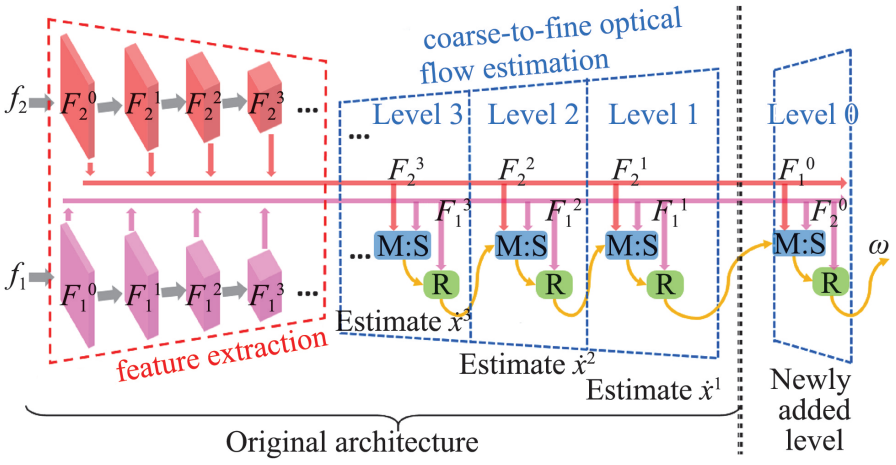

FIG. 1: Schematic diagram of the improved CNN network structure

(1) *Feature deformation*: In order to solve the velocity field estimation problem of large displacement, Hui et al. (2018) propose to reduce the distance between features F_1 and F_2 by feature warping (i.e., distortion), with F_2 passing the predicted velocity of the previous layer $\dot{x}$. Compared with the image deformation method used in the correlation analysis method WIDIM or multiscale variational optical flow solution, the feature-based deformation can make the network more accurate and more efficient.

(2) *Feature matching and subpixel refinement* (M:S module in Fig. 1): Pixel matching by calculating the correlation between feature F_1 and deformed feature F_2, resulting in a rough velocity vector estimation, however, the correlation matching only achieves maximum pixel-level precision. Therefore, the estimated velocity field needs to be refined to subpixel precision by minimizing the feature space distance.

(3) *Velocity field regularization* (R module in Fig. 1). In order to smooth the estimated velocity field and avoid the occurrence of outliers, the proposed network uses feature-driven local convolution operations to achieve velocity field regularization. The regular term in the split optical flow objective function is similar. However, the regularization function of the proposed network is adaptively learned for the local characteristics of the optical flow field, and its convolution kernel is obtained through training.

The proposed network training uses the following loss function:

$$\varepsilon_{Loss} = \sum_i \lambda_i e_i \ . \tag{2}$$

It consists of output errors of different layers with different weighting coefficients. Among them, i denotes different levels, e_i denotes the error metric between the predicted velocity field and the real velocity field, and λ_i denotes the weight of different levels.

In order to make it more suitable for complex flow field motion estimation, the network structure and training parameters of the proposed network need to be modified. The improved ideas include: first, add the reverse convolutional layer extraction velocity field in the decoder part, replacing the last in the original network structure. The interpolation operation of one layer is shown in the newly added part of Fig. 1. Second is to redistribute the weight of the loss function and increase the proportion of the layer 0 network. Third is to re-adjust the parameters of the data normalization operation.

2. PIV DATA SET GENERATION AND NETWORK TRAINING

2.1 Generating PIV Data Sets

The proposed flow estimation algorithm needs to adopt the supervised learning strategy. However, training the neural network through supervised learning requires real value data to optimize the model parameters. Unfortunately, PIV experiments often

fail to obtain accurate velocity fields, therefore a synthetic data sets are needed to perform CNN training. According to the general method of experimental fluid mechanics, we first generate the particle image and the flow velocity field, and then move the position of the particle symmetrically through the motion field to obtain the image pair, see the example in Fig. 2. Every single data item contains an image pair (input) and a true velocity field (output). In Step 1 and Step 2, a particle image and a flow motion pattern are generated. In Step 3, the positions of the particles are shifted symmetrically by the flow motion to generate an image pair and true theoretical result.

(1) *Generate particle images*. To generate particle images, a particle image generator can be used, assuming that a particle can be described by a two-dimensional Gaussian function:

$$I(x, y) = I_0 \exp\left[\frac{-(x - x_0)^2 - (y - y_0)^2}{\frac{1}{8} d_p^2}\right], \tag{3}$$

where I_0 is the peak intensity in the Gaussian center, d_p denotes the particle diameter, and (x_0, y_0) is the position of the particle. Note that I_0, d_p, and (x_0, y_0) can be diverse from each particle. Furthermore, let us define the particle seeding density of the image as ρ (unit: particle per pixel, ppp), which is a global parameter for an image and affects the particle number in the observed domain. With specified values of I_0, d_p, and (x_0, y_0), a typical particle image can be produced. Parameters for defining an image can be randomly selected in a proper range, as shown in Table 1. Figure 3 displays three examples of particle images produced by different parameters. All the images are in the resolution of 256 × 256 pixels.

(2) *Generate flow velocity field*. Computational fluid dynamics (CFD) is used to artificially generate the flow velocity field. In order to increase the diversity of dataset, various flow modes are used in this paper. CFD can simply simulate some standard flow fields, such as uniformity, flow field, reverse step flow field, and cylindrical flow field (hereinafter referred to as uniform, back-step, cylinder, respectively). In addition, this paper obtains more fluid motion data directly from some open source literature, such as Carlier (2005) and Resseguier et al. (2007), respectively provide the same free

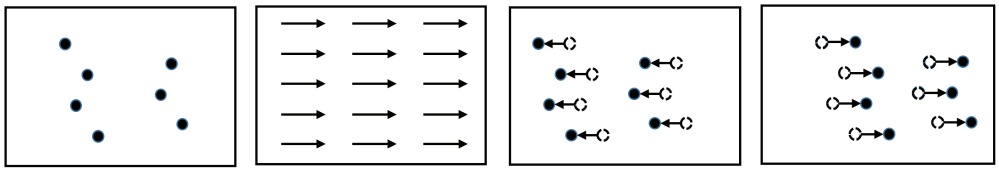

FIG. 2: Principle of generating dataset for training of the deep neural network for PIV

TABLE 1: Ranges of parameters for generating particle image

Parameter	Range	Unit
I_0	200–255	Particle per pixel
d_p	1–4	Pixel
ρ	0.05–0.1	Gray value

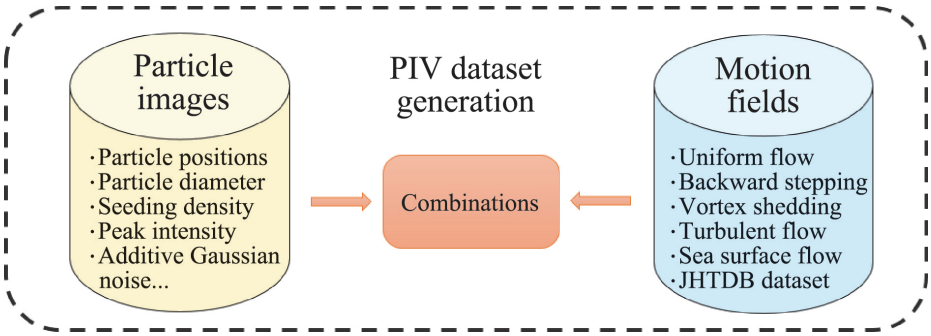

FIG. 3: Illustration of the PIV data set generation

turbulent flow field (referred to as DNS turbulence) and the surface flow field (surface quasi geostrophic, SQG) simulation model. In addition, the Johns Hopkins turbulence database (JHTBD) (Li et al., 2008) also provides a variety of turbulent velocity field data. Table 2 gives a list of the types of fluid motion velocity fields used, where the diversity of flow fields can be increased by adjusting different velocity magnitudes and different Reynolds numbers. For example, for cylinder flow simulation, flow fields at different Reynolds numbers (Re = 40, 150, 200, 300, and 400) can be used to enhance the range of training data, to ensure that the algorithm is applicable to different vortex shedding flows.

TABLE 2: Description of the motion fields for neural network training

Case Name	Description
Uniform	Uniform flow in the whole image domain
Back-step	Backward stepping flow
Cylinder	Vortex shedding flow over a circular cylinder
DNS-turbulence	A homogeneous and isotropic turbulent flow
SQG	Sea surface flow driven by a surface quasi-geostrophic model
JHTDB-channel	Channel flow provided by Johns Hopkins turbulence databases

By randomly combining the particle image generated above and the flow velocity field, a PIV data set can be formed, as shown in Fig. 2. For this paper more than 13,000 particle image pairs and velocity field true values were used, involving 6 different flow field experiments, more than 10 different operating conditions (such as different Reynolds numbers), with an average of about 1000 data samples per operating condition. By randomly combining the particle image generated above with the flow velocity field, an initial CNN model with trained parameters is obtained.

As mentioned above, real life PIV experiments often obtain imperfect and noisy images full, as compared to ideal images used for CFD calculations. The system trained purely based on CFD often fails to produce accurate results when experimental and imperfect image pairs are used as input. In order to address it, a second step of training the CNN model is performed by adding noise during particle image generation and changing the brightness of the image based on Gaussian curve.

The particle size in the first image is randomly selected according to the uniform distribution between 1 and 5 pixels, and the particle concentration is randomly selected according to the uniform distribution between 0.01 and 0.19. The particle size of the second image is changed compared to the first image, and the variation range is randomly selected according to the uniform distribution between –20% and 20%. Then each row or column of the image is multiplied by a Gaussian curve, and the row and column are random.

The initial results of the calculations obtained after adding this extra training layer, showed a very promising improvement as compared to system trained only with perfect images, and we will continue adding more similarly generated data set to improve the accuracy of the CNN model. Further detailed studies are needed to assess the level of improvements.

2.2 Network Training

Once the structure of the convolutional neural network is determined and there is a data set available for training, the parameters of the network can be trained by defining a minimized loss function (for convolution operations, the unknown parameter is the convolution kernel). As mentioned, the network structure used is shown in Fig. 1. The data set is generated by Fig. 4. The objective function of the network optimization is given in Eq. (2). Since the training data set is usually large, in order to speed up the training efficiency, higher efficiency can be adopted. The gradient-based optimization algorithm, such as the long gradient algorithm in the convolutional neural network, uses the stochastic Adam gradient optimization algorithm. After 1.2 × 106 iterations, the network parameters tend to converge. The trained network model is called deep PIV. Subsequent experimental tests are generated using this training.

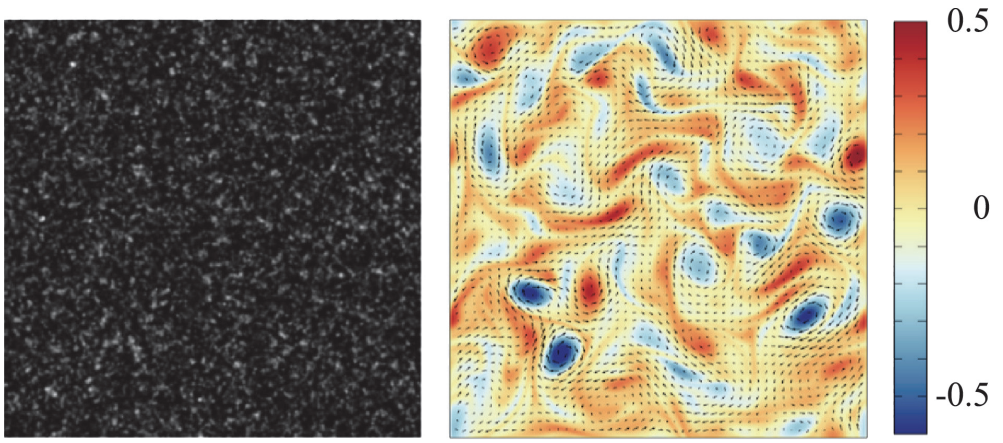

FIG. 4: Particle image of DNS turbulent flow and the corresponding velocity field with vorticity (t = 50)

3. SIMULATION IMAGE TEST RESULTS AND DISCUSSION

3.1 Introduction to Turbulent Particle Image

In order to verify the accuracy and reliability of the complex fluid motion estimation algorithm, a simulation data set with real values is needed for corresponding testing. This chapter cites the two-dimensional turbulent flow particle image given in Carlier (2005). The image sequence describes direct numerical simulation (DNS) particle motion of two-dimensional turbulent flow field, experimental Reynolds number Re = 3000, Schmidt number Sc = 0.7. The image sequence contains 100 samples, and the image size is 256 × 256 pixels. Figure 5 shows the particle image at t = 50 and

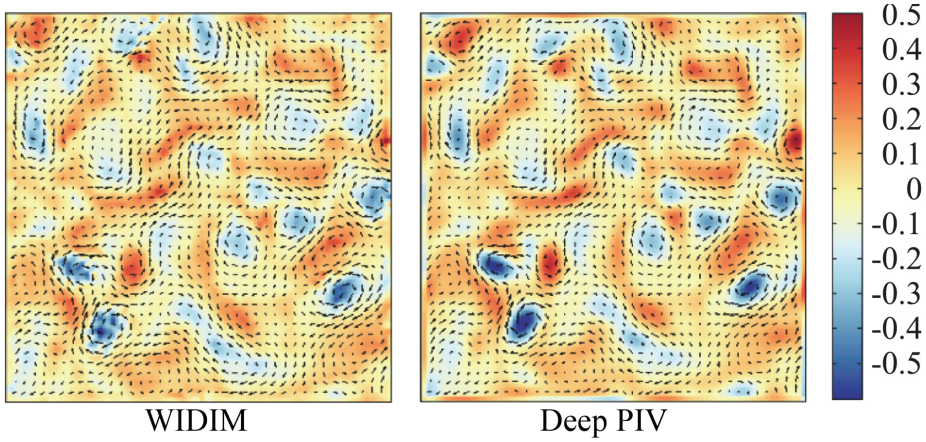

FIG. 5: Velocity fields and vorticity maps of DNS turbulent flow provided by WIDIM and deep PIV (t = 50)

the corresponding real velocity field. The DNS turbulent particle image dataset is an internationally accepted benchmark for PIV algorithm testing.

When there is a real value as a reference, the most commonly used PIV algorithm is the root mean square error (RMSE), which is calculated as follows:

$$\delta_{RMSE}=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left[(u_i^t-u_i^e)^2+(v_i^t-v_i^e)^2\right]}, \tag{4}$$

where (u^t, v^t) and (u^e, v^e) represent the true velocity vector and the velocity vector estimated by the algorithm, respectively, i refers to all velocity vector numbers. After acquiring the velocity field information, the corresponding vorticity can be easily calculated from the velocity field. In this way the algorithm's estimation of the vortex structure can be judged.

The proposed deep neural network is compared with the two more mature and representative image velocimetry algorithms described in the introduction: WIDIM, and the optical flow method (Horn et al., 1981) with multiscale pyramid. Among them, WIDIM algorithm has low computational complexity and high robustness, but due to limited estimation resolution, the estimation of small-scale velocity field of turbulent flow is not good enough. The optical flow algorithm has high calculation accuracy and can provide dense velocity field (down to maximum single pixel level), but it requires more computation time due to variational optimization. Using deep neural network to achieve dense velocity field estimation will be accurate, high resolution, and achieve near real-time calculation speed.

3.2 Test Results and Discussion

Figure 5 shows the velocity field and vorticity map estimated by the WIDIM algorithm and the deep PIV model at t = 50 in the turbulent particle image sequence. As can be seen from Fig. 5, the vorticity calculated by the WIDIM algorithm is not continuous enough. The values differ greatly, and the velocity field and vorticity map calculated by the neural network model are basically consistent with the true values. Especially in the estimation of small-scale vortex structures, the advantage of the deep PIV neural network model is more obvious, which is due to the algorithm. It can provide a highly accurate dense velocity field.

The root mean square error curves of the three different algorithms are shown in Fig. 6. As shown in Fig. 6, the performance of the WIDIM method is much worse overall than the multiscale optical flow method, showing that the correlation analysis based on window matching is small and the vortex structure estimation is insufficient. Deep PIV is superior to the variational optical flow method, and the RMSE is the smallest in the entire image sequence. This experiment shows that the use and improvement based on the convolutional neural network is very effective for PIV motion estimation, especially in the turbulent flow (including small-scale vortex structures).

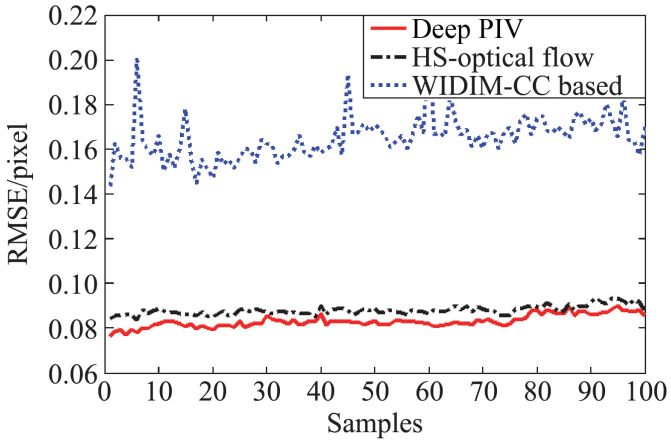

FIG. 6: RMSE error of different algorithms for turbulent particle image sequences

Table 3 also shows the time when different algorithms perform one calculation in the same experiment. The system configuration used in the experiment is Intel Core i7 G7700 CPU 3.60 GHz and NVIDIA GTX 1080Ti GPU. Graphics processing unit (GPU) speeds up the execution of CNN model. Efficiency has obvious advantages. Even if it is converted to CPU operation mode, the calculation time of CNN model is less than that of WIDIM method. It should be noted that the deep PIV proposed in this paper can extract maximum single pixel level from particle image, just like HS variable wavelength optical flow. Velocity field, but the method based on deep neural network is much more efficient than the HS method. In summary, in the two-dimensional turbulent particle image test, the first proposed deep learning motion estimation model is superior in accuracy and efficiency as compared to traditional correlation analysis and optical flow method.

In order to study the working principle of the PIV neural network more deeply, the turbulent particle image (t = 50) is processed by deep PIV as an example to analyze the output result of the network intermediate layer. As mentioned above, in the network structure (Fig. 1), the encoder function is to extract feature pyramids, while decoder implements coarse-to-fine velocity field estimation for these features. Deep PIV uses network layering and the coarse-to-fine computational concept is very simi-

TABLE 3: Computational time of different methods for the DNS turbulent flow image pair

Method	t/ms	Number of Vectors
WIDIM (CPU)	509	61 × 61
HS (CPU)	2294	256 × 256
Deep PIV (GPU)	47	256 × 256

lar to the multiscale pyramid iterative HS optical flow algorithm. There are three main differences between them:

(1) when constructing a pyramid, the HS optical flow method is to down sample the image, while CNN model is to extract features while sampling;

(2) in decoder calculation in the velocity field, the CNN estimator operates in the above automatically extracted feature map;

(3) all parameters of the filter (convolution kernel) in the neural network are trained from the data, and the HS method depends on the artificial selection and settings.

Figure 7 shows the characteristic output of the first two layers of the encoder subnetwork under the trained convolution of the input particle image, corresponding to the F^0 and F^1 modules of Fig. 1. According to Fig. 7, the trained convolution kernel is anisotropic filtering operations on particle images (including smoothing, sharpening, particle stretching, particle magnification, reduction, etc.) to form different texture features. Compared to the original particle image *f*, there are more texture features. The image plays a beneficial role in correlation matching and velocity field calculation in the decoder algorithm.

Figure 8 shows the estimated velocity fields for different pyramid levels in the decoder subnetwork, corresponding to Level 3 to Level 0 layers in Fig. 1, and com-

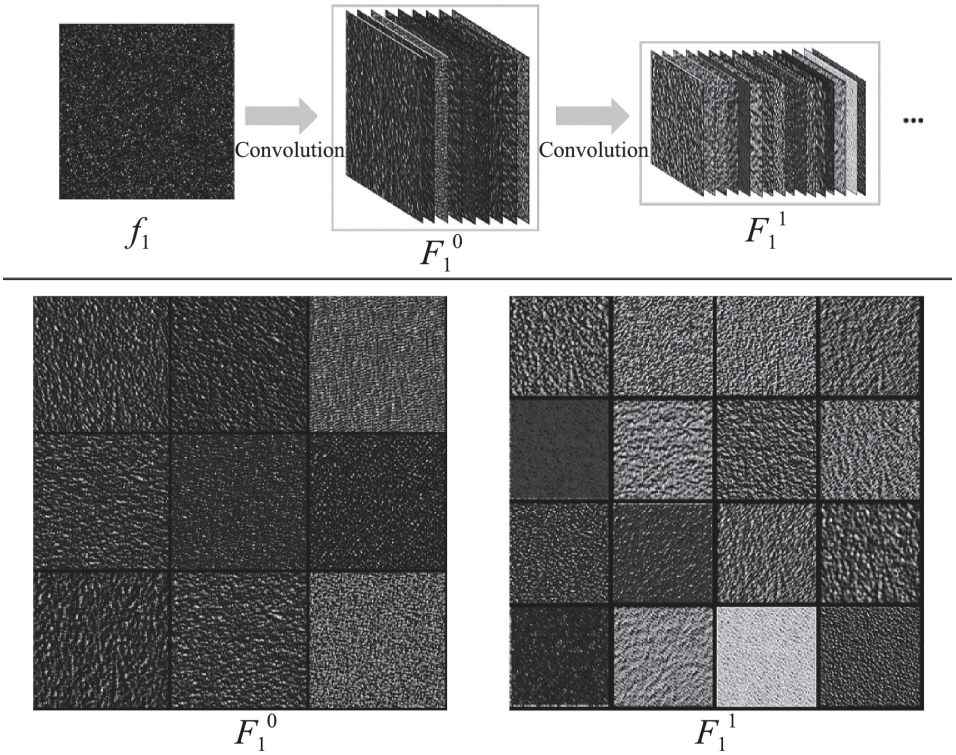

FIG. 7: Characteristics of deep PIV extracted particle images

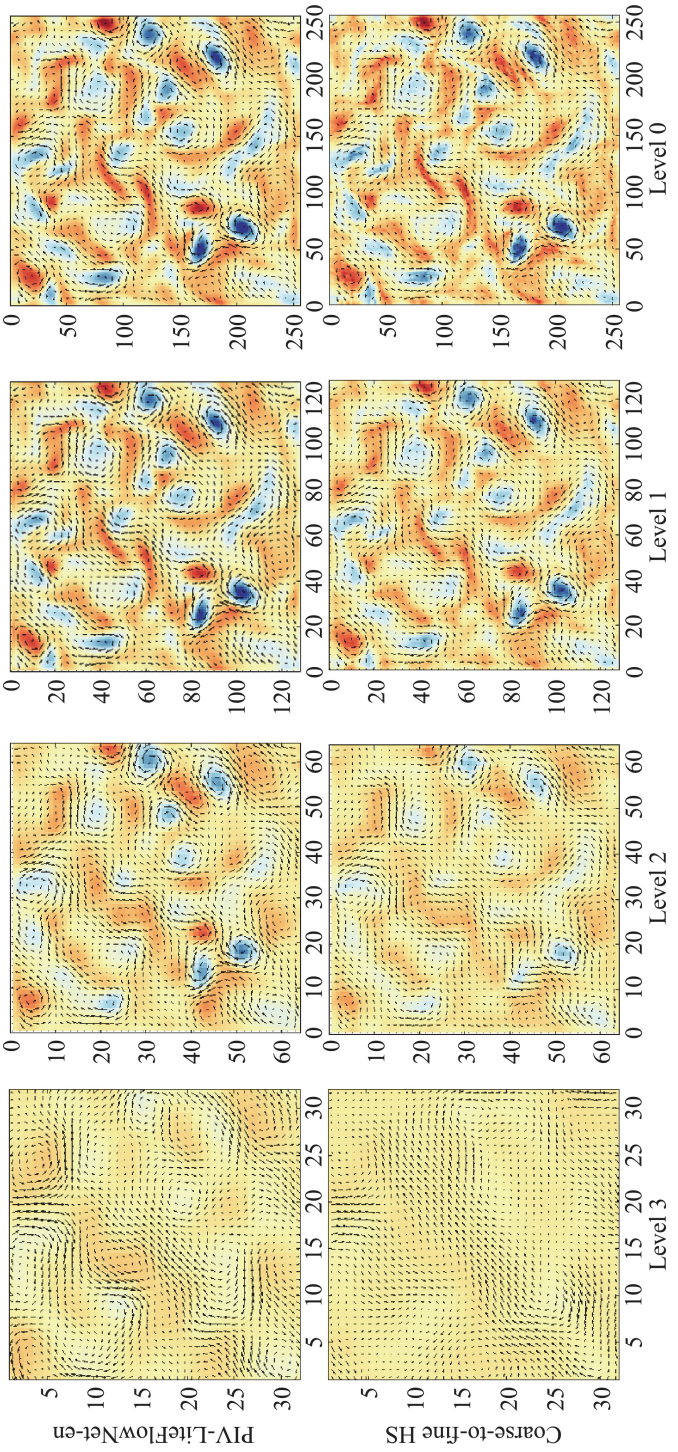

FIG. 8: Outputs at different resolution levels of deep PIV and coarse-to-fine HS method

pared with the multilayer HS optical flow method. As can be seen from Fig. 8, comparing any level of the output of deep PIV is more reasonable than the multiscale HS method. The reason for this difference may be that the deep PIV network uses image features for calculation, while the HS method directly uses down sampled particle images. At low resolution, the estimated velocity field on the image will be used for the next level of deformation and so on, so it will also affect the final output. The above is an analysis of the internal output and intrinsic principles of the deep PIV neural network.

4. FLOW OVER AN AEROFOIL PIV EXPERIMENT RESULTS AND DISCUSSION

In order to demonstrate the practicality of the particle image velocimetry algorithm based on deep neural network, deep PIV was tested with real PIV experimental data conducted in a wind tunnel.

The flow over an aerofoil experiment was in an open circuit wind tunnel available at the Department of Aerospace Engineering, Indian Institute of Technology, Kanpur, India. The tunnel has a 3000-mm-long test section with a square cross section of 610 mm × 610 mm. The tunnel has a square three-dimensional contraction section with 16:1 contraction ratio.

4.1 Experimental Method and Basic Setup

A two-dimensional PIV technique was utilized to measure the instantaneous velocity fields in the spanwise or cross-flow plane over the flying wing model, as schematically shown in Fig. 9. The flow is seeded with the help of a commercial fog generator (Antari Z1500II-R) with Antari FLG fog liquid. The fog generator was placed at the diffusor section of the closed-circuit wind tunnel. A dual head Nd–YAG laser (Quantel Evergreen, 200 mJ/pulse, 15 Hz) was used for illuminating the flow field of the seeding particles. A CCD camera (Imperx) with 8 MP resolution (3212 pixels × 2488 pixels) was used for capturing the illuminated particles. The laser and the camera were synchronized using a synchronizer (procured from IDTvision, USA). The camera field of view was 135 mm × 101 mm. The PIV data acquisition was carried out using the MicroVec software (procured from Vision Asia, Singapore). The PIV measurements were carried out at 30% of the root chord location at the freestream velocity, 30 m/s. We may mention that the measurements were carried out at the cross-flow plane which perpendicular to the model surface, as seen in Fig. 9. However, two images are acquired, and these images were analyzed using Microvec 3.6.1 software. The software utilizes a traditional WIDIM cross-correlation algorithm and 16 pixel × 16 pixel correlation window is used.

Figure 10 shows the estimated velocity vectors for the two algorithms in the jet aerofoil experiment with interrogation window of 16 × 16 pixels for WIDIM and

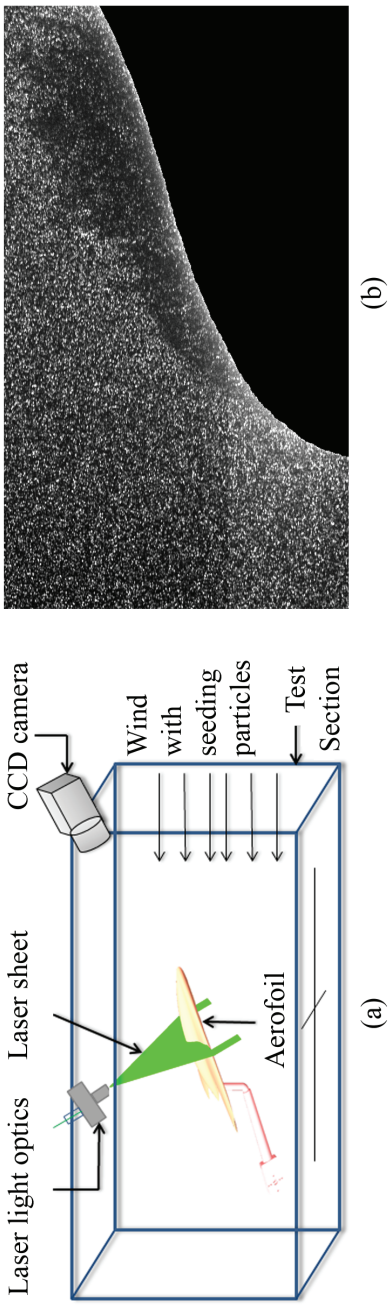

FIG. 9: Schematic of the particle image velocimetry experimental setup (a), and its particle image with mask (b)

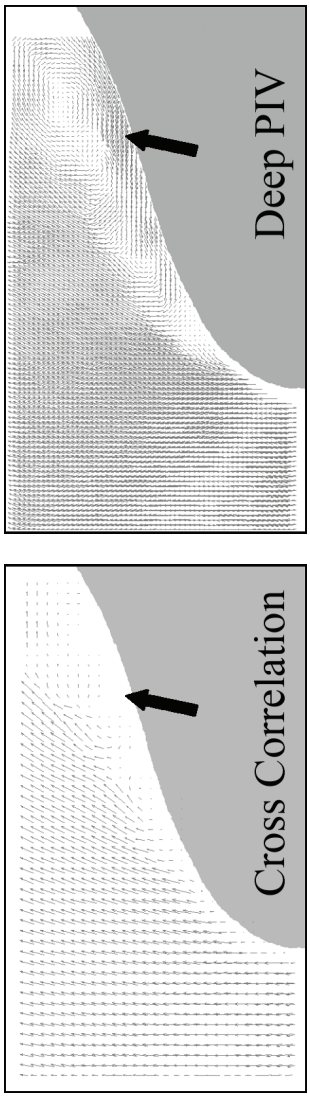

FIG. 10: The estimation of two algorithms in the aerofoil PIV experiment

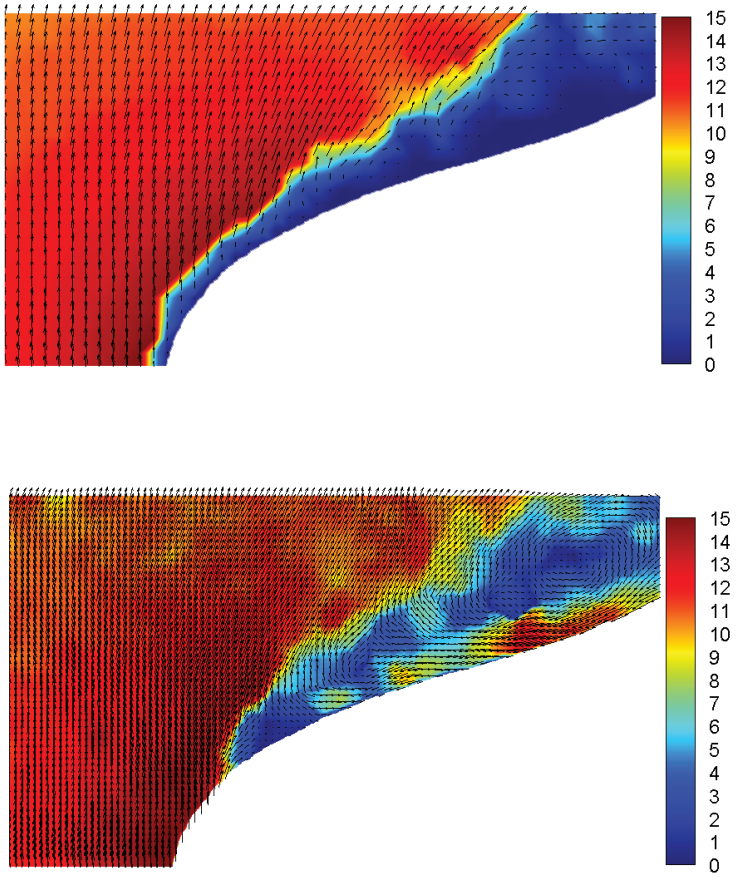

FIG. 11: Estimated velocity field amplitude map of aerofoil flow

spatial resolution of maximum 1 × 1 pixel for deep PIV. Figure 11 shows the velocity amplitude map for the velocity field. The results from both calculations are shown next to each other for comparison.

In most regions, the vector of deep PIV is identical with that of the cross-correlation WIDIM PIV. This implies that both deep PIV and traditional PIV can resolve the large-scale flow structures very accurately. The greatest difference appears in the region indicated by the red arrow, where the back flow is very strong due to the low pressure. The back flow causes high velocity gradient in this small region. Therefore, the resolution of the traditional PIV is insufficient to extract the small scales in this region. However, as shown in this figure, the deep PIV can resolve the back flow very well. It shows that the spacial resolution of deep PIV can achieve much more accurate results than the traditional PIV. It should be noted that the real PIV image used in the experiment is not included in the data set used in CNN network training. Therefore,

the experimental results also show that the deep neural network trained by artificially generating PIV data can also be successfully applied in real PIV experiments to obtain dense velocity field information.

5. CONCLUSIONS

This paper attempts to study the particle image velocimetry with deep neural network, and combines artificial intelligence with traditional experimental fluid mechanics. Firstly, the optical flow neural network from computer vision is used for motion estimation. Based on its structure is based and with parameter improvement; subsequently, synthetic PIV particle image datasets are used for supervised training of neural networks; finally, a deep neural network model suitable for fluid motion estimation is obtained. This network model can provide velocity fields down to maximum single-pixel level resolution efficiently and accurately.

The preliminary experimental evaluation of the turbulent flow field particle image is carried out. The test results show that the deep PIV model has advantages in accuracy, resolution and computational efficiency compared with the traditional correlation analysis method and optical flow method. At the same time, the hidden layer of PIV neural network is discussed. The output and the intrinsic principle are used for subsequent research. Finally, the algorithm is tested in the real jet PIV experiment. The result of deep PIV is equal and, in some instances, even better than commercial PIV software based on cross correlation.

It is verified that the particle image velocimetry algorithm based on deep neural network is practical and has wide application prospects.

REFERENCES

Adrian, L., Ronald, J.A., and Westerweel, J., *Particle Image Velocimetry*, no. 30, Cambridge, UK: Cambridge University Press.

Cai, S., Liang, J., Zhou, S., Gao, Q., Xu, C., Wei, R., Wereley, S., and Kwon, J.S., Deep-PIV: A New Framework of PIV using Deep Learning Techniques, Paper presented at *13th Int. Symp. on Particle Image Velocimetry*, Munich, Germany, 2019a.

Cai, S., Zhou, S., and Xu, C., Dense Motion Estimation of Particle Images via a Convolutional Neural Network, *Exp. Fluids,* vol. **60**, no. 4, pp. 60–73, 2019b.

Carlier, J., Second Set of Fluid Mechanics Image Sequences, *European Project Fluid Image Analysis and Description (FLUID)-http://www. fluid.irisa.fr*, 2005.

Chen, X., Zille, P., Shao, L., and Corpetti, T., Optical Flow for Incompressible Turbulence Motion Estimation, *Exp. Fluids,* vol. **56**, no. 1, pp. 1–14, 2015.

Corpetti, T., Heitz, D., Cansino, A.G., Mémin, E., and Santa Cruz, A., Fluid Experimental Flow Estimation Based on an Optical-Flow Scheme, *Exp. Fluids,* vol. **40**, no. 1, pp. 80–97, 2006.

Heitz, D., Mémin, E., and Schnörr, C., Variational Fluid Flow Measurements from Image Sequences: Synopsis and Perspectives, *Exp. Fluids,* vol. **48**, no. 3, pp. 369–393, 2010.

Horn, B.K.P. and Schunck, B.G., Determining Optical Flow, *Artificial Intelligence,* vol **17**, nos. 1–3, pp. 185–203, 1981.

Hui, T.W., Tang, X., and Chen, C.L., Liteflownet: A Lightweight Convolutional Neural Network for Optical Flow Estimation, *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, 2018.

Krizhevsky, A., Sutskever, I., and Hinton, G.E., Imagenet Classification with Deep Convolutional Neural Networks, *Adv. Neural Information Process. Syst*., vol. **25**, no. 2, pp. 1097–1105, 2012.

LeCun, Y., Bengio, Y., and Hinton, G., Deep Learning, *Nature,* vol. **521**, pp. 436–444, 2015.

Lee, Y., Yang, H., and Yin, Z., PIV-DCNN: Cascaded Deep Convolutional Neural Networks for Particle Image Velocimetry, *Exp. Fluids,* vol. **58**, no. 12, p. 171, 2017.

Li, Y., Perlman, E., Wan, M., Yang, Y., Meneveau, C., Burns, R.C., Chen, S., Szalay, A.S., and Eyink, G., A Public Turbulence Database Cluster and Applications to Study Lagrangian Evolution of Velocity Increments in Turbulence, *J. Turbulence,* vol. **9**, no. 31, pp. 1–29, 2008.

Liu, T. and Shen, L., Fluid Flow and Optical Flow, *J. Fluid Mech.,* vol. **614**, pp. 253–291, 2008.

Marusic, I., Large Spatial Range Measurements in High Reynolds Number Wall-Bounded Flows, Paper presented at *13th Int. Symp. on Particle Image Velocimetry*, Munich, Germany, 2019.

Rabault, J., Kolaas, J., and Jensen, A., Performing Particle Image Velocimetry using Artificial Neural Networks: A Proof-of-Concept, *Meas. Sci. Technol.,* vol. **28**, no. 12, p. 125301, 2017.

Raffel, M., Willert, C.E., Wereley, S., and Kompenhans, J., *Particle Image Velocimetry: A Practical Guide,* Springer, 2018.

Resseguier, V., Mémin, E., and Chapron, B., Geophysical Flows under Location Uncertainty, Part II: Quasi-Geostrophy and Efficient Ensemble Spreading, *Geophys. Astrophys. Fluid Dyn.,* vol. **111**, no. 3, pp. 177–208, 2017.

Ruhnau, P., Kohlberger T., Schnörr, C., and Nobach, H., Variational Optical Flow Estimation for Particle Image Velocimetry, *Exp. Fluids,* vol. **38**, no. 1, pp. 21–32, 2005.

Scarano, F., Iterative Image Deformation Methods in PIV, *Meas. Sci. Technol.,* vol. **13**, no. 1, pp. 1–19, 2002.