

# Unsupervised Learning of Particle Image Velocimetry

Mingrui Zhang and Matthew D. Piggott

Department of Earth Science and Engineering,  
Imperial College London, South Kensington Campus, London, UK

**Abstract.** Particle Image Velocimetry (PIV) is a classical flow estimation problem which is widely considered and utilised, especially as a diagnostic tool in experimental fluid dynamics and the remote sensing of environmental flows. Recently, the development of deep learning based methods has inspired new approaches to tackle the PIV problem. These supervised learning based methods are driven by large volumes of data with ground truth training information. However, it is difficult to collect reliable ground truth data in large-scale, real-world scenarios. Although synthetic datasets can be used as alternatives, the gap between the training set-ups and real-world scenarios limits applicability. We present here what we believe to be the first work which takes an unsupervised learning based approach to tackle PIV problems. The proposed approach is inspired by classic optical flow methods. Instead of using ground truth data, we make use of photometric loss between two consecutive image frames, consistency loss in bidirectional flow estimates and spatial smoothness loss to construct the total unsupervised loss function. The approach shows significant potential and advantages for fluid flow estimation. Results presented here demonstrate that our method outperforms competitive results compared with classical PIV methods as well as supervised learning based methods for a broad PIV dataset, and even outperforms these existing approaches in some difficult flow cases.

**Keywords:** Particle Image Velocimetry (PIV) · Velocity field diagnostics · Deep learning · Unsupervised learning

## 1 Introduction

Particle Image Velocimetry (PIV) is one of the most popular measurement techniques in experimental fluid dynamics, and is also used to diagnose flow information from the remote sensing of large-scale environmental flows. The method provides quantitative measurements of velocity fields in fluids that can be used to explore complex flow phenomena. When conducting the PIV technique, the fluid under investigation is seeded with sufficiently small tracer particles (or the presence of naturally occurring features is exploited). These particles are assumed to follow the flow dynamics. With illumination (in the laboratory often through the use of lasers to capture image information over a two-dimensional plane), the particles in the fluid are visible. By comparing resulting flow images between

time levels, velocity field information can be inferred [1]. There are two main techniques used for performing classical PIV: cross-correlation and variational optical flow methods.

The development of deep learning techniques has inspired a new direction for tackling PIV-like problems. Several authors have in the literature proposed and demonstrated the use of supervised learning based methods for PIV. However, due to the unavailability of a broad range of reliable ground truth training data, supervised learning methods have limitations, especially when seeking to generalise to real-world problems. On the other hand, unsupervised learning is a type of machine learning approach that looks for previously undetected patterns in a dataset with no pre-existing labels and with minimum human supervision [2].

In this paper we propose a new fluid velocity estimation method using an unsupervised learning strategy based upon particle images.

### 1.1 Cross-correlation and variational optical flow methods

There are two main standard approaches for performing particle image velocimetry: cross-correlation and optical flow methods. The cross-correlation method calculates a displacement by searching for the maximum cross-correlation between two interrogation windows from an image pair [3], e.g. such as in the *WIDIM* (window deformation iterative multi-grid) method. The cross-correlation method is efficient and relatively easy to implement. However, it only outputs a spatially sparse (compared to the resolution of the seed particles in the fluid) displacement field and requires post-processing. The variational optical flow method was proposed by Horn and Schunck (HS) [4]. It is a motion estimation approach that has been applied to PIV problems [5]. It treats the PIV problem through the solution of an optimisation problem, seeking the minimisation of an objective function. The method can output a dense displacement field, but the optimisation process is time-consuming.

### 1.2 Deep learning methods

Machine learning methods, especially deep learning, have made great progress in applications to many real-world problems in recent years. In the PIV community, deep learning has been introduced recently. In [6], the authors provided a proof-of-concept on this topic, where artificial neural networks are designed to perform end-to-end PIV for the first time in this work.

PIV techniques are closely related to computational photography, a sub-domain of computer vision. In this community, there are several important works related to the motion estimation problem using deep learning. The *FlowNetS* and *FlowNetC* networks [7] were the first proposed for dense optical flow estimation. *FlowNet2* [8], an extension of *FlowNet*, improves the optical flow estimation to a state-of-the-art level. In addition, a lighter-weight network *LiteFlowNet* [9] has also been proposed. It achieves a similar level accuracy compared to *FlowNet2*, using less trainable parameters. Although the networks mentioned above have

achieved excellent performance for estimating motion fields from consecutive image pairs, their applications is generally limited to rigid or quasi-rigid motion.

Therefore, it is of interest to explore the performance of these existing networks on particle image velocimetry problems.

## 2 Related work

Supervised and unsupervised learning are two different learning approaches. The key difference is that supervised learning requires ground truth data while unsupervised learning does not.

### 2.1 Supervised learning methods

End-to-end supervised learning using neural networks for PIV was first introduced by Rabault et al. in [6]. A convolutional neural network and a fully-connected neural network were trained to perform PIV on several test cases. That work provided a proof-of-concept for the research community. However, the trained model did not achieve the ultimate quality of result compared with traditional PIV methods, and the application scenarios considered were limited to relatively simple cases. Lee et al. [10] proposed a cascaded network architecture. The network was verified to produce results comparable to standard PIV methods (one-pass cross-correlation method, three-pass window deformation). However, it had larger computational costs and lower efficiency. Another deep architecture approach based on supervised learning was proposed by Cai et al. in [11]. In that work the author developed a motion estimator *PIV-FlowNetS-en* based upon *FlowNet*. The estimator is able to extract features from particle images and output dense displacement fields. The model was evaluated both on synthetic and experimental data, and was shown to achieve good accuracy with high efficiency compared to correlation-based PIV methods such as the *WIDIM* method. Follow-up work introduced a more complex but lighter-weight network *PIV-LiteFlowNet-en* [12], based on *LiteFlowNet* [9]. The model was shown to have the same level of performance as variational optical flow methods in terms of estimation accuracy, while showing advantages in terms of efficiency.

The supervised learning approach relies heavily on large volumes of training data. However, in most real-world scenarios, especially in fluid dynamics, there is no easily available ground truth data and/or it is extremely difficult to annotate the data accurately through human means. Although the use of synthetic data (e.g. based upon computational fluid dynamics studies) can help construct large annotated datasets, the gap between synthetic and real-world scenarios limits the generalisation abilities of the constructed networks. This can mean that supervised learning based approaches may struggle when confronted with data from real-world problems.

## 2.2 Unsupervised learning methods

Unsupervised learning is a type of machine learning that, in contrast, looks for previously undetected patterns in a dataset with no pre-existing labels and with minimum human supervision.

To the best of our knowledge, there are no previous examples of approaches that tackle the PIV problem based on unsupervised learning. In the computer vision community, there is some previous work related to the use of unsupervised learning for optical flow estimation. Yu et al. [13] suggested an unsupervised method based on *FlowNet* in order to sidestep the limitations of synthetic datasets. They use a loss function that combines a data term (photometric loss) measuring photometric constancy over time with a spatial term (smoothness loss) modelling the expected variation of flow across the image. In [14], Meister et al. extended the work using a symmetric, occlusion-aware loss based on both forward and backward optical flow estimates. They also made use of iterative refinement by stacking multiple *FlowNet* networks. The model showed advantages and outperformed supervised learning on challenging realistic benchmarks.

Our work is inspired in part by Meister et al. [14]; we extend the unsupervised learning strategy to PIV problems, building our model based on *LiteFlowNet* instead of *FlowNet*. We trained our model on a synthetic PIV dataset generated by Cai et al. in [11]. Unlike the supervised strategy, we only use the particle images pairs in the dataset, and leave the ground truth motion data (which is used to generate the image pairs) for benchmarking purposes.

## 3 Method

Given a grayscale image pair  $I_1, I_2 : P \rightarrow \mathbb{R}^1$  as input, our goal is to estimate the forward flow field from  $I_1$  to  $I_2$ ,  $\mathbf{F}^f \equiv (u^f, v^f)^T$ , where  $u_f$  and  $v_f$  are scalar velocity fields in two orthogonal directions. As we take the bidirectional estimate into consideration, the backward flow field is defined as  $\mathbf{F}^b \equiv (u^b, v^b)^T$ . In section 3.1 we will introduce the unsupervised loss and how the loss is integrated for training. The network architecture will be described in section 3.2.

### 3.1 Unsupervised Loss

In the training process, the input only contains image pairs  $I_1, I_2$ , without the velocity field ground truth. Therefore, we use traditional optical flow measurements to evaluate our results. The total unsupervised loss is a combination of photometric loss, estimate flow smoothness loss and consistency loss between forward and backward fields.

**Photometric loss.** The photometric loss is defined in terms of the difference between the first image and the warped second image using the forward flow field estimate, and the difference between the second image and the warped first

image using the backward estimate. The bidirectional photometric loss is thus defined as the sum of two parts:

$$L_P(I_1, I_2, \mathbf{F}^f, \mathbf{F}^b) = \sum_{\mathbf{x} \in P} \rho(I_1(\mathbf{x}) - I_2(\mathbf{x} + \mathbf{F}^f(\mathbf{x}))) + \rho(I_2(\mathbf{x}) - I_1(\mathbf{x} + \mathbf{F}^b(\mathbf{x}))), \quad (1)$$

where  $\rho(\cdot)$  is the generalized Charbonnier penalty function,  $\rho = (x^2 + \epsilon^2)^\gamma$ , which is a differentiable, robust convex function [15]. We use the values  $\gamma = 0.45$ ,  $\epsilon = 10^{-3}$  in this work.

Image ‘backwarping’ is the key step when computing the photometric loss. In order to make the loss backpropagation possible during the training process, we use the differentiable bilinear sampling scheme proposed in [16]. The basic idea is first to generate a sampling coordinate in target image  $I_2$ , using  $I_1$  and the flow field estimate  $\mathbf{F}^f$ . The coordinate can be described as:  $\mathbf{x}_s = \mathbf{x} + \mathbf{F}^f(\mathbf{x}) = (x_1 + F_u^f, y_1 + F_v^f)$ , here  $\mathbf{x}$  is the coordinate field for image  $I_1$ . A bilinear sampler is then used to construct the warped image in terms of coordinate  $\mathbf{x}$ :

$$I_{\text{warp}}(\mathbf{x}) = \sum_{x_s^i, y_s^i \in \mathbf{x}_s} I_2(\mathbf{x}_s) \max(0, 1 - |x_s - x_s^i|) \max(0, 1 - |y_s - y_s^i|). \quad (2)$$

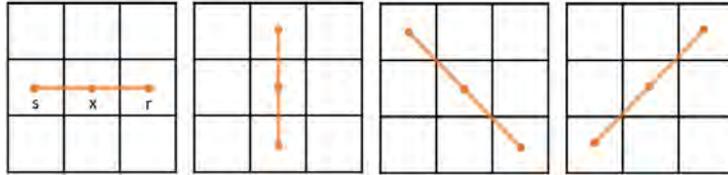
**Smoothness loss.** There are regions in the images that lack necessary information. For example, there may be insufficient particles near image boundaries, as the particles move out of the image area in the second frame or the particles have not entered the image in the first frame. Therefore, to tackle resulting ambiguities, a smoothness loss is included into our total unsupervised loss. To enhance the regularisation effects, we use a second-order smooth constraint [17]:

$$L_S(\mathbf{F}^f, \mathbf{F}^b) = \sum_{(\mathbf{s}, \mathbf{r}) \in \mathbf{N}(\mathbf{x})} \sum_{\mathbf{x} \in P} \rho(\mathbf{F}^f(\mathbf{s}) - 2\mathbf{F}^f(\mathbf{x}) + \mathbf{F}^f(\mathbf{r})) + \rho(\mathbf{F}^b(\mathbf{s}) - 2\mathbf{F}^b(\mathbf{x}) + \mathbf{F}^b(\mathbf{r})), \quad (3)$$

where  $\mathbf{N}$  represents a four channel filter ( $x$ ,  $y$  and two diagonals, see Fig. 1). Therefore, the process here is first to compute the convolution of the two flow components ( $u$  in the  $x$  and  $v$  in the  $y$  directions) with the four channel filter respectively, then compute their Charbonnier loss.

**Consistency loss.** The forward and backward flow estimates should be consistent, i.e. the forward flow  $\mathbf{F}^f$  is expected to be the inverse of the backward flow  $\mathbf{F}^b(\mathbf{x} + \mathbf{F}^f)$  at the corresponding pixel in the second image. The sum of this pair of flow fields should therefore be zero, and similarly for the backward flow estimate. The consistency loss function can thus be defined as:

$$L_C(\mathbf{F}^f, \mathbf{F}^b) = \sum_{\mathbf{x} \in P} \rho(\mathbf{F}^f + \mathbf{F}^b(\mathbf{x} + \mathbf{F}^f)) + \rho(\mathbf{F}^b + \mathbf{F}^f(\mathbf{x} + \mathbf{F}^b)). \quad (4)$$



**Fig. 1.** Four channel filter used in the smoothness loss: in the directions  $x$ ,  $y$  and the two diagonals, shown in the frames above from left to right respectively.  $s$ ,  $x$ ,  $r$  indicate the three neighboring pixels considered for each direction.

**Final integrated loss.** The final integrated loss,  $L$ , combines the above loss terms using weighted (with scalar weights  $\lambda_D$ ,  $\lambda_S$ ,  $\lambda_C$ ) summation:

$$L(I_1, I_2, \mathbf{F}^f, \mathbf{F}^b) = \lambda_D L_D + \lambda_S L_S + \lambda_C L_C. \quad (5)$$

### 3.2 Network architecture

**UnLiteFlowNet-PIV.** Our network, named *UnLiteFlowNet-PIV*, is based on *LiteFlowNet* [9]. It extracts two images' features using a two-stream convolution neural network (NetC) with shared weights. NetC has a pyramidal structure and encodes the image from full resolution to a sixth of that of the original. Then a decoder (NetE) performs cascaded flow inference (convolutionally upsampling) with flow regularisation. The final flow estimate is upsampled to the original resolution using bilinear interpolation. In our work, we compute both the forward and backward flow in one estimation. The input for the forward flow estimation is  $(I_1, I_2)$ , and it is  $(I_2, I_1)$  for the backward flow.

**Training Loss.** The training loss function's design is similar to that in *FlowNet* [7] and *LiteFlowNet* [9], and uses a multi-scale resolution loss. It is the weighted sum of the estimation losses from each of the intermediate layers:

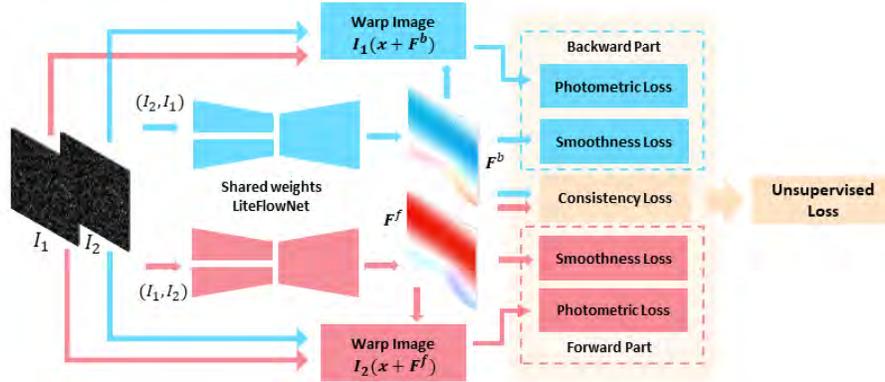
$$L_T = \sum_i w_i L_i, \quad (6)$$

where  $L_i$  is the loss function (5). At each layer, the image pair  $(I_1, I_2)$  is down-sampled to compute the current layer's loss. As the distance between pixels effectively changes after downsampling, the flow estimate is multiplied by the appropriate scaling factor, which is the fraction between the current and the full image resolution. Here,  $i_{\max} = 6$ , and  $L_6$  indicates the loss at full resolution.

## 4 Evaluation

### 4.1 PIV dataset

The dataset considered in this work was generated by Cai et al. [11]. The dataset contains 15,050 particle image pairs with the originating flow field ground truth



**Fig. 2.** Data flow for *UnLiteFlowNet-PIV*, from inputs to unsupervised loss. Due to taking bidirectional flows into consideration, the red components represent the forward part, with the image pair  $(I_1, I_2)$  as input. The blue components indicate the backward part. Although there are two networks shown for clarity, since shared weights are used there is only one in the implementation.

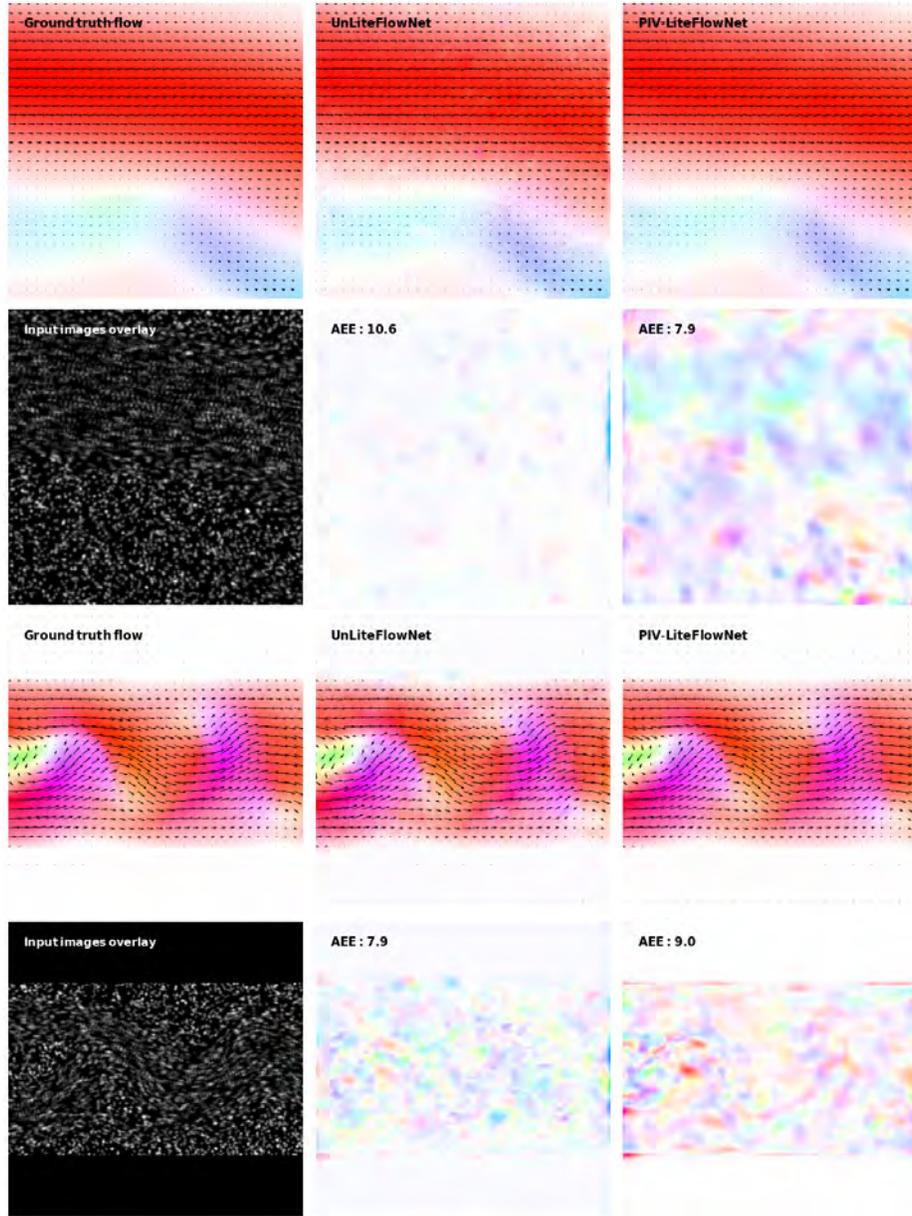
data obtained from computational fluid dynamics simulations. There are eight different types of flow contained in the dataset, including ‘uniform’ flow, flow past a backward facing step (‘back-step’) and past a ‘cylinder’, both at a variety of Reynolds numbers, ‘DNS-turbulence’, sea surface flow driven by a quasi-geostrophic model (‘SQG’), etc. Detailed information on the dataset is provided in Table 2. In our work we use half of the dataset for training and the other half for testing.

## 4.2 Training details

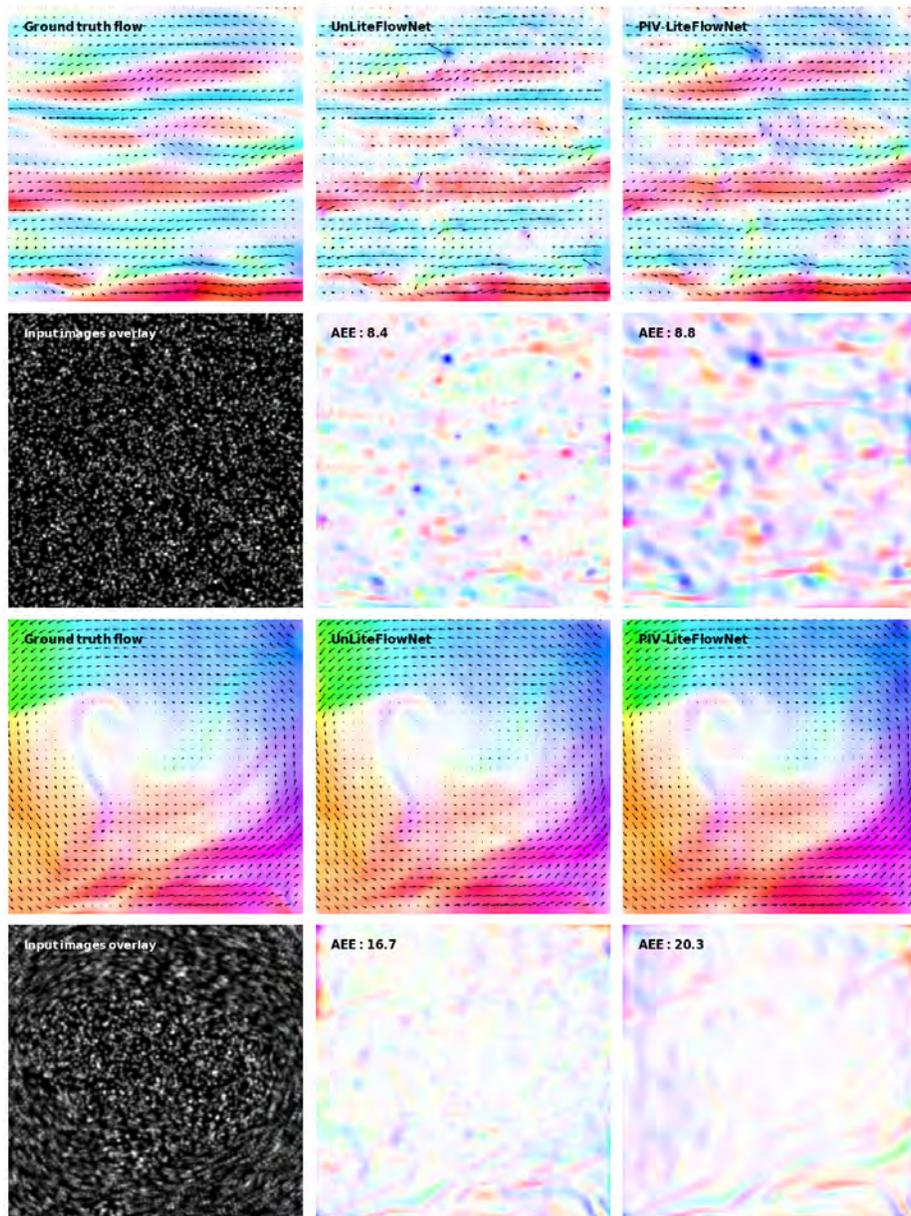
We train the model for 40,000 iterations with a batch size of four image pairs using the Adam optimiser. The learning rate is kept at  $10^{-4}$ . The smoothness loss weight is  $\lambda_C = 3.0$ , the consistency loss weight  $\lambda_C = 0.2$ , photometric loss weight  $\lambda_D = 1.0$ . The weights for different layers are set to [12.7, 5.5, 4.35, 3.9, 3.4, 1.1] as in [14], from the full resolution to the lowest level. The image pair are normalized from value ranges of 0–255 to 0–1 before feeding into the network.

## 4.3 Results

Table 1 compares the accuracy of our model with previous work and different approaches, including classical PIV and state-of-the-art deep learning based methods. The results are evaluated on the PIV dataset, with the Averaged Endpoint Error (AEE) calculated for different flow types. In order to compare the results easier, we set the units of the AEE to pixel per 100 pixels. The AEE can be described as the  $L^2$ -norm of the difference in flow estimation  $\mathbf{F}^e$  and the flow ground truth  $\mathbf{F}^g$ :



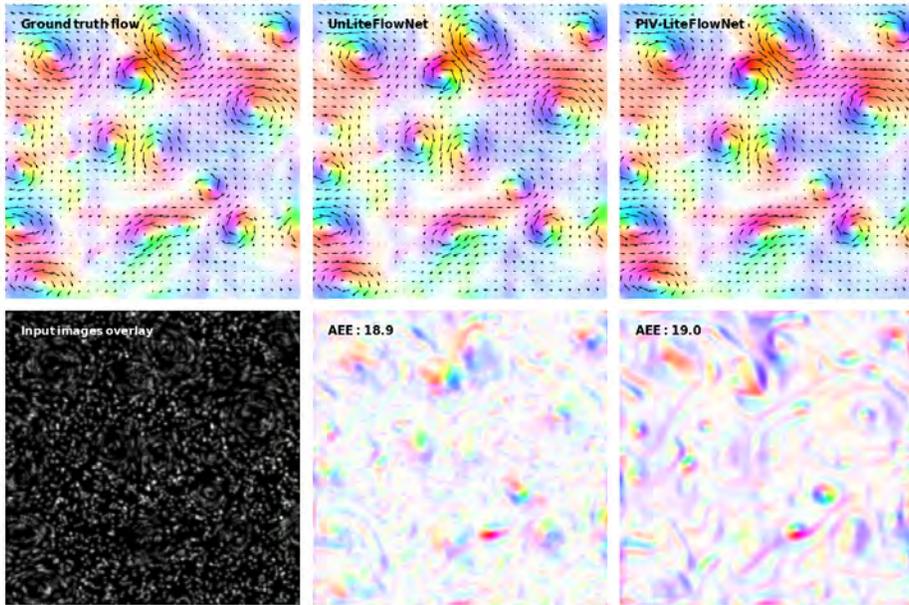
**Fig. 3.** Visual comparisons between ground truth flow data (left), our fully unsupervised model *UnLiteFlowNet-PIV* (middle) and *PIV-LiteFlowNet* (right) shown on the first and third rows for respectively the ‘Back-step’ and ‘Cylinder’ flow cases. The flow field colour is coded in HSV [18]. The second and fourth rows show the input images overlays (left), along with the errors and corresponding Averaged Endpoint Error (AEE) values for the two networks and image pairs considered. In the error plots the white colour indicates zero error, and pixel colour with higher saturation represents larger errors. It can be observed that even though our new unsupervised model never has access to the ground truth during training, it still tends to outperform the supervised model. See also Figs. 4 and 5 for further cases.



**Fig. 4.** Caption as for Fig. 3 but now for the flow cases ‘JHTDB-channel’ (first two rows) and ‘DNS-turbulence’ (final two rows). It can be observed that errors are distributed along the vortex boundaries in both models. However, our *UnLiteFlowNet-PIV* model does tend to demonstrate lower errors in the interior, e.g. around the regions with strong internal vortices.

$$\text{AEE} = \|\mathbf{F}^e - \mathbf{F}^g\|_2. \quad (7)$$

**Comparison to classical PIV.** It can be observed that our unsupervised model outperforms classical correlation-based PIV WIDIM methods in almost all flow cases, especially for the challenging cases of DNS-turbulence and SQG. Although the unsupervised model does not outperform the Horn–Schunck (HS) optical flow method [4], the differences are relatively small. In addition, as mentioned above, the HS optical flow method requires a large amount of computational time in order to conduct the optimisation process, which results in low efficiency especially when multiple image pairs need to be processed. Without considering the time to load images from disk, the computational time for 500 image ( $256 \times 256$ ) pairs using our *UnLiteFlowNet-PIV* is 10.17 seconds on an Nvidia Tesla P100 GPU, while the HS optical method requires roughly 556.5 seconds and WIDIM (with a window size of  $29 \times 29$ ) requires 211.5 seconds on an Intel Core I7-7700 CPU [12]. Although the classical PIV methods are tested on a CPU, as shown in the [6, 11] the speed improvements for them running on GPUs are limited. Therefore, efficiency is a great advantage for learning based methods compared to the classical approaches.



**Fig. 5.** Caption similar to Fig. 3, but now only showing the ‘SQG’ case. The *UnLiteFlowNet-PIV* model again shows more than competitive performance on this challenging flow case.

**Comparison to deep learning PIV.** The unsupervised learning approach shows potentially significant advantages compared to state-of-the-art supervised learning methods. Figs. 3, 4 and 5 demonstrate comparisons between our fully unsupervised model *UnLiteFlowNet-PIV* and *PIV-LiteFlowNet*. *PIV-LiteFlowNet* [12] uses a similar network architecture to our *UnLiteFlowNet-PIV*, but is trained using a supervised learning strategy with ground truth data. Although the unsupervised *UnLiteFlowNet-PIV* never has access to the ground truth data during the training process, it still outperforms most supervised learning methods (*PIV-NetS-noRef*, *PIV-NetS*, *PIV-LiteFlowNet*), especially on difficult cases. Therefore, the unsupervised learning method with an accurate loss function shows competitive capabilities and often better performance compared to supervised methods.

*PIV-LiteFlowNet-en* [12] is an enhanced version of *PIV-LiteFlowNet*, it adds one additional layer at the end of the NetE, which improves its inference ability but makes the network more complicated and heavier. We did not try to construct deeper networks in our work for brevity. There are ideas for improving the performance by stacking networks [8], which would also be an interesting avenue to explore in further work.

**Table 1.** Averaged Endpoint Error (AEE) for the PIV dataset (averaged over all image pairs), the error unit is set to pixel per 100 pixels for easier comparison. From top to bottom, WIDIM and HS optical flow are the classical PIV methods described in section 1.1, the next four rows are state-of-the-art supervised learning methods described in section 2.1. The final row shows results of our unsupervised method introduced in this work.

Methods	Back-Step		Cylinder		JHTDB channel		DNS turbulence		SQG	
	train	test	train	test	train	test	train	test	train	test
WIDIM [12]	-	3.4	-	8.3	-	8.4	-	30.4	-	45.7
HS optical flow [12]	-	4.5	-	7.0	-	6.9	-	13.5	-	15.6
PIV-NetS-noRef [11]	13.6	13.9	19.8	19.4	24.6	24.7	50.6	52.5	51.9	52.5
PIV-NetS [11]	5.8	5.9	6.9	7.2	16.3	15.5	27.1	28.2	28.9	29.4
PIV-LiteFlowNet [12]	5.5	5.6	8.7	8.3	10.9	10.4	18.8	19.6	19.8	20.2
PIV-LiteFlowNet-en [12]	3.2	3.3	5.2	4.9	7.9	7.5	11.6	12.2	12.4	12.6
<b>UnLiteFlowNet-PIV</b>	-	10.1	-	7.8	-	9.6	-	13.5	-	19.7

## 5 Conclusion

We present here the first work using an unsupervised learning approach for solving Particle Image Velocimetry (PIV) problems. The proposed unsupervised

learning approach shows significant promise and potential advantages for fluid flow estimation. It yields competitive results compared with classical PIV methods as well as existing supervised learning based methods, and even outperforms them on some difficult flow cases. Furthermore, the unsupervised learning method does not rely on any ground truth data in order to train, which makes it extremely promising to generalize to complex real-world flow scenarios where ground truth is effectively unknowable, and thus represents a key advantage over supervised methods.

**Table 2.** Detailed description of the PIV dataset considered, from [11].  $dx$  refers to the particle displacements considered between two image frames in units of number of pixels. Re refers to the Reynolds numbers considered. ‘JHTDB’ implies that the data was taken from the Johns Hopkins turbulence databases [19]. Refer to [11] for further details.

Metric Name	Description	Condition	Quantity
Uniform	Uniform flow	$ dx  \in [0, 5]$	1000
Back-step	Flow past a backward facing step	Re = 800	600
		Re = 1000	600
		Re = 1200	1000
		Re = 1500	1000
Cylinder	Flow past a circular cylinder	Re = 40	50
		Re = 150	500
		Re = 200	500
		Re = 300	500
		Re = 400	500
DNS-turbulence	Homogeneous and isotropic turbulent flow	-	2000
SQG	Sea surface flow driven by SQG model	-	1500
Channel flow	Channel flow provided by JHTDB	-	1600
JHTDB-mhd1024	Forced MHD turbulence provided by JHTDB	-	800
JHTDB-isotropic1024	Forced isotropic turbulence provided by JHTDB	-	2000

### Acknowledgements

The authors would like to acknowledge funding from the Chinese Scholarship Council and Imperial College London (a pump priming research award from the

Energy Futures Lab, Data Science Institute and Gratham Institute – Climate Change and the Environment) that supported this work.

## References

1. R. Adrian and J. Westerweel.: Particle Image Velocimetry. Cambridge University Press, (2011)
2. G. Hinton and T. Sejnowski.: Unsupervised Learning: Foundations of Neural Computation. MIT Press, (1999)
3. J. Westerweel.: Fundamentals of digital particle image velocimetry. *Experiments in Fluids*, vol. 23, no. 12, pp. 1379–1392 (1997)
4. B. Horn and B. Schunck.: Determining optical flow. *Artificial Intelligence*, vol. 17, no. 1–3, pp. 185–203 (1981)
5. P. Ruhnau, T. Kohlberger, C. Schnorr, and H. Nobach.: Variational optical flow estimation for particle image velocimetry. *Experiments in Fluids*, vol. 38, no. 1, pp. 21–32 (2005)
6. J. Rabault, J. Kolaas, and A. Jensen.: Performing particle image velocimetry using artificial neural networks: a proof-of-concept. *Measurement Science and Technology*, vol. 28, no. 12, p. 125301 (2017)
7. A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van der Smagt, D. Cremers, and T. Brox.: FlowNet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2758–2766 (2015)
8. E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox.: FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, (2017)
9. T. Hui, X. Tang, and C. Loy.: LiteFlowNet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8981–8989 (2018)
10. Y. Lee, H. Yang, and Z. Yin.: PIV-DCNN: cascaded deep convolutional neural networks for particle image velocimetry. *Experiments in Fluids*, vol. 58, no. 12, article number 171 (2017)
11. S. Cai, S. Zhou, C. Xu, and Q. Gao.: Dense motion estimation of particle images via a convolutional neural network. *Experiments in Fluids*, vol. 60, no. 4, article number 73 (2019)
12. S. Cai, J. Liang, Q. Gao, C. Xu, R. Wei.: Particle image velocimetry based on a deep learning motion estimator. *IEEE Transactions on Instrumentation and Measurement* (2019b)
13. J. Yu, A. Harley, and K. Derpanis.: Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *European Conference on Computer Vision ECCV 2016*, pp. 3–10 (2016)
14. S. Meister, J. Hur, and S. Roth.: UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. *The Thirty-Second AAAI Conference on Artificial Intelligence* (2018)
15. D. Sun, S. Roth, and M. Black.: A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, vol. 106, no. 2, pp. 115–137 (2014)
16. M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu.: Spatial transformer networks. In *NIPS’15: Proceedings of the 28th International Conference on Neural Information Processing Systems – Volume 2*, pp. 2017–2025 (2015)

17. C. Zhang, Z. Li, R. Cai, H. Chao, and Y. Rui.: As-rigid-as-possible stereo under second order smoothness priors. In European Conference on Computer Vision ECCV 2014, pp. 112–126 (2014)
18. S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski.: A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, vol. 92, no. 1, pp. 1–31 (2011)
19. Y. Li, E. Perlman, M. Wan, Y. Yang, C. Meneveau, R. Burns, S. Chen, A. Szalay, and G. Eyink.: A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, vol. 9, no. 9, (2008)