

Neural Network Applications for Particle Image Velocimetry Analysis of Steady Fluid Flow: A Proof of Concept

Sashwat Prasadh^{1*†}, Rebecca Ralph^{2†}, Joseph Fitzgerald^{1†},
Jeanette Wheeler^{2†}

¹Department of Physics & Physical Oceanography, Memorial University of Newfoundland, 283 Prince Philip Drive, St. John's, A1B 3P7, NL, Canada.

²Department of Biology, Memorial University of Newfoundland , 45 Arctic Ave, St. John's, A1C 5S7, NL, Canada.

*Corresponding author(s). E-mail(s): sprasadh@mun.ca;

†All authors contributed equally to this work.

Abstract

Particle Image Velocimetry (PIV) is an experimental technique to obtain an instantaneous, whole-field, averaged velocity field. The standard cross-correlation algorithm used within is mediocre at best, failing to take into account any fluid dynamics equations or statistics. We test FeedForward and Convolutional Neural Networks(NN) as an alternative using steady synthetic velocity fields of known statistical properties. This is studied as a function of Noise and Gradient intensities in train and test sets. The results show that cross-correlation algorithms are typically better at predicting the velocity field. However, introducing copious amount of noise in the training set enables CNN's to predict more accurately on datasets with high intensity noise and gradients. The study is further extended to test inclination of NN to predict on the statistic that they were trained on. The results indicate that NN do not have a preference for any statistic included in training. Moreover, it was observed that NN hardly suffered from unwanted small perturbations in the velocity field compared to Cross-correlation. The home-brewed PIV code is publicly available on [G](#)

Keywords: Particle Image Velocimetry, Neural Network, Gaussian Noise

1 Introduction

Flow visualizations and measurements of velocity fields are vital for a vast range of applications extending far beyond traditional fluid mechanics. In this regard, Particle Image Velocimetry (hereafter PIV) is a prominent and widely used fluid flow measurement technique (See [Raffel et al, 2018](#)). The method works by measuring the displacement between two successive snapshots of tracer particles that follow the fluid flow to determine the average velocity and visualize the fluid flow vector field. In principle, such an idea of flow visualization was suggested in the earlier part of the 20th century ([Reynolds, 1883](#); [Prandtl, 1905, 1936](#); [Willert and Kompenhans, 2010](#)). The current form of PIV started in 1980's ([Bastiaans, 1993](#)) and significantly advanced with the advent of charge-coupled devices. [Keane and Adrian \(1992\)](#) proposed replacing auto-correlation with cross-correlation (hereafter ccr) which has become the standard today. Further advancements have followed in forms of pre-processing and post processing (See [Stamhuis and Thielicke, 2014](#)). We refer [Adrian \(2005\)](#) for a brief history, [Hassan et al \(2020\)](#) for applications, [Scharnowski and Kähler \(2020\)](#) for planar PIV, and [Rohacs et al \(2023\)](#) for a comprehensive review.

PIV has an edge over other technique such as Hot Wire Anemometry (HWA) and Particle Tracking Velocimetry (PTV) due to the nature of its non-invasive, whole field, and instantaneous averaged velocity measurements ([Raffel et al, 2018](#); [Johnson, 2016](#)). Challenges persist in the forms of resolution, high velocities and gradients, and uncertainty quantification. To address these, many advancements such as [Adatrao et al \(2021\)](#) have followed over the years. To improve resolution, optical flow has been widely utilized, but this highly suffers from factors such as noise. PIV has been adapted into more specialized methods such as time resolved ([Beresh, 2021](#)), utilizing multiple cameras called stereoscopic PIV ([Prasad, 2000](#)), micro ([Etminan et al, 2022](#); [Lindken et al, 2006](#)), three-dimensional such as tomographic PIV ([Scarano, 2013](#)), among other forms. See the cited works for purpose and limitations. Collectively, these methods are still far from ideal, given the vitality of PIV to studies ranging from medicine ([G. Yazdi et al, 2020](#)) to engineering, the need for improvement remains ever-present.

Machine learning and AI have shown tremendous potential over the last few years in the tasks of fluid mechanics. [Mendez et al \(2022\)](#) provides a comprehensive review and [Viquerat et al \(2022\)](#) provides a timeline for specific sub field applications of reinforcement deep learning.

These advancements include applications of artificial intelligence to PIV, particularly neural network(s) (hereby NN(s)). In fact, NNs had been suggested for PIV application within a decade of digitization of particle image velocimetry, for instance, [Grant and Pan \(1993\)](#); [Carosone et al \(1995\)](#); [Grant and Pan \(1997\)](#); [Hassan and Philip \(1997\)](#) but likely the computation resources were lacking at the time. The interest was reignited with [Rabault et al \(2017\)](#) who used Fully connected and Convolutional Neural Networks for a proof concept test. The results showed that CNN are more accurate than fully connected networks in most PIV tasks, and both of them are comparable to, but not yet surpassing a state of the art PIV cross correlation (ccr) software. Similarly, [Lee et al \(2017\)](#) used a cascaded CNN to test a range of flows and the results showed promise. Further advancements, for both specific characteristic and

overall improvements have followed since (See [Discetti and Liu, 2023](#); [Discetti and Sciacchitano, 2023](#); [Yu et al, 2023](#), for a thorough review). Such a comparison between cross-correlation and AI PIV (using CNN's and deep learning) has extended to even study CFD blood flow simulations pertaining to aneurysms ([Majewski et al, 2021](#)). Furthermore, Physics Informed Neural Networks (PINN) have also shown success in aiding PIV measurements, for instance [Eivazi et al \(2024\)](#).

The experimental setup design for measuring PIV remains without doubt an excellent construct for measuring an average velocity. However, the cross correlation algorithm that essentially extracts the vector field from the images is limited to statistical principles. Thereby, not accounting for the underlying governing equations such as Navier stokes and continuity equations and rendering them obsolete within the analysis. The potential for PINN's to fully replace cross correlation remains in questions as the loss function is entirely dependant on a Navier Stokes solver, and therefore the PINN is restricted to its limitations as well

As a starting point, the purpose of this work is to investigate the potential of Deep Feedforward and Convolutional Neural network(s) (Hereafter FNN(s) and CNN(s)) against the traditional PIV cross correlation on simple translational synthetic velocity flows. In theory, cross correlation algorithms work assuming lateral translation of particles without any divergence, noise or any other such factors. This study is then extended to random flows.

We further investigate the implication of Gaussian white noise on the accuracies of the PIV, CNN and FNN. The motivating principle behind this is that PIV is widely used for biological oceanography to construct velocity ([Wheeler et al, 2013](#)) and pressure ([Calicchia et al, 2023](#)) fields around marine pelagic life. Noise resulting from the movement of the object is widely present in such experiments that is not accounted for by cross correlation algorithms. We aim to test whether introducing noise during NNs training tunes the model to better predict on similar data.

Similarly, PIV is a primary technique in physical oceanography to study stratified/Internal flow ([Luzzatto-Fegiz, 2022](#); [Seelig et al, 2019](#)), Ice sheet modelling([Johnson, 2020](#)), and Turbulent Flows ([McCutchan and Johnson, 2020](#); [Augier et al, 2017](#)). Turbulent flows are ubiquitous to nature and best studied using statistics. Among other techniques PIV can be used to measure vorticity or determine the nine components of a deformation rate tense/Reynolds Stress Tensor using a Probability Density Function (See the latter for a review [Nieuwstadt et al, 2016](#); [Westerweel et al, 2013](#)). These technique are characteristic to specific parameters that determine the nature of these systems. To this end, we investigate whether a NN trained on data that stems from certain statistical properties show any preference to predicting on that same statistical character.

Therefore, based on the aforementioned motivations, the purpose of this work is to conduct an overall study on FNN and CNN for PIV with random steady flows, test the effects of noise for training and testing sets, and investigate whether the NN show any preference for predicting on a certain statistical characteristic that they are trained on. The three studies are compared to a home brewed traditional cross correlation PIV code.

The structure of this paper is as follows.

2 Method

Neural networks require large datasets to accurately map input values to their corresponding true values. The precision of these true values significantly influences model performance. Experimentally, acquiring such extensive Particle Image Velocimetry (PIV) datasets is costly, and even if obtained, the accuracy of the predictions remains constrained by the limitations of the cross-correlation algorithm, neglecting additional influencing factors.

To address this, we generate synthetic PIV images, which can be produced in large quantities. The custom-developed code for synthetic data generation and cross-correlation PIV analysis is available at [Q](#). Computations are performed on an RTX 4070-Ti, achieving a generation rate of approximately 625 one-dimensional snapshot pairs per minute. Since high-resolution two-dimensional image generation is computationally expensive, we present a one-dimensional framework that preserves the essential characteristics of a conventional PIV setup. The methodology and implementation details are discussed in the following sections.

2.1 Synthetic data set & PIV

Neural networks process large datasets iteratively in each epoch to identify patterns. To prevent overfitting, it is necessary to introduce a degree of randomization to the flow, ensuring the network does not learn unintended recurring patterns or redundant fields. To achieve this, we generate periodic random steady flows with predefined statistical properties. This is accomplished by multiplying a spectrally decomposed covariance matrix by Gaussian white noise, enforcing a correlation between elements based on their separation distance.

The covariance matrix initially contains negative eigenvalues and complex eigenvectors due to limited numerical precision. To rectify this, we retain only the real eigenvectors and set negative eigenvalues to zero. This approach follows [Penland \(1996\)](#), who further scaled positive eigenvalues to improve covariance matrix reconstruction. However, since the difference between the reconstructed and original covariance matrices is negligible ([Figure 11](#)), we omit scaling. The covariance matrix is sampled from a Gaussian distribution ([Gauss, 1809](#); [Stahl, 2006](#)):

$$Cov_Matrix = A_0 \exp\left(\frac{-(x - x_0)}{2\sigma^2}\right) \quad (1)$$

Where A_0 is the amplitude of the curve. A known limitation of cross-correlation-based PIV is its sensitivity to high velocities; larger displacement increases the likelihood of spurious vectors. While it remains unclear whether the same principle applies to neural networks, it is reasonable to expect similar behavior. To constrain PIV displacements, we set $A_0 = 0.1$, ensuring that particle displacement adheres to the one-quarter rule, which states that the maximum displacement should not exceed one-quarter of the interrogation window ([Johnson, 2016](#)). In our setup, this limits displacement to 8 pixels in a 1024-pixel vector ([Figure 3](#)). The corresponding 2D velocity field and divergence plot are also presented.

This setup allows control over velocity variation by adjusting the full width at half maximum (FWHM), denoted as σ , of the Gaussian. Except in section 4, where σ varies from 0.001 to 0.1 in increments of 0.01 for testing, we fix $\sigma = 0.1$. This variation enables an assessment of whether neural networks exhibit a preference for the statistical properties of their training data.

For particle seeding, simulations indicate that random seeding at moderate densities results in interrogation windows with insufficient or absent particles, leading to poor cross-correlation and erroneous vectors. To maintain a random yet homogeneous distribution, we seed ten particles per interrogation window, achieving an average density of 0.3 particles per pixel (0.001 ppp in 2D), consistent with literature values. [Sciacchitano et al \(2013\)](#); [Lagemann et al \(2022\)](#) show that increasing seeding density reduces RMS error. Since density changes with dimensionality, direct comparisons with literature are not feasible, but our chosen density ensures a strong correlation signal.

Particles are interpolated from the grid, fitted to a line, and advected one step backward ($t-dt$) and forward ($t+dt$) using a fourth-order Runge-Kutta integrator ([Runge, 1895](#); [Kutta, 1901](#); [Butcher, 1996](#)). The resulting advected particles are converted into an intensity field using

$$I(x) = I_0 \exp\left(\frac{-(x - x_0)^2}{d_\tau^2/8}\right) \quad (2)$$

Where d_τ is the particle diameter, and is selected as 2 pixels to avoid the well known peak locking effect ([Nogueira et al, 2021, 2011](#); [Michaelis et al, 2016](#)). Monte Carlo simulations ([Raffel et al, 2018](#)) and analytical studies ([Cameron, 2022](#)) also recommend diameters of 2 pixels or larger to minimize camera noise effects. The peak intensity I_0 set to 250, consistent with literature ([Cai et al, 2019](#); [Yu et al, 2023](#)). High-intensity stable points may form due to particle clustering, mimicking experimental speckle patterns ([Kawahashi and Hirahara, 2000](#)), which can introduce spurious vectors. To counteract this, intensity values are capped at 250.

This procedure generates two synthetic PIV snapshots. In a laboratory setting, equivalent data would include background noise, gradients, out-of-plane particles, and other disturbances. To analyze their effects, we create four snapshot variants:

1. Silent: A clean, ideal version
2. Noise: Includes artificial Gaussian white noise.
3. Gradient: Includes a random intensity gradient.
4. Noise & Gradient: Combines both effects.

[Figure 1](#) illustrates these variations. The corresponding 2D intensity distributions are shown in [Figure 12](#). Noise and gradient intensities are separately randomly sampled within 0–40% of the maximum intensity. The gradient’s center is randomized to prevent the network from learning fixed locations. Lastly, vectors are downsampled from 4096 to 1024 pixels to simulate CCD sensor properties. This step will be referred to as pixelization. A comparison between experimental and synthetic images is provided in [Figure 13](#).

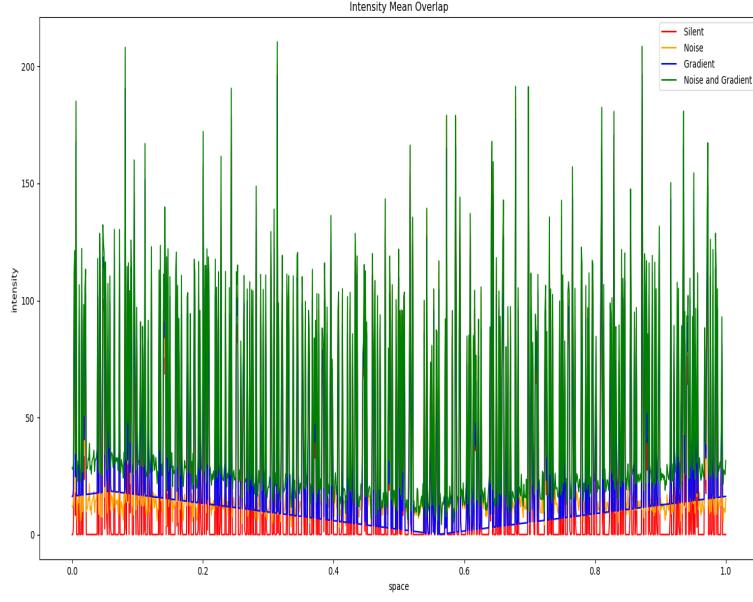


Fig. 1 Caption

Cross-correlation on these snapshots follows equation 3(Raffel et al, 2018)-

$$R(r) = \sum_{i=0}^m \sum_{r=-n}^n IA_1(i).IA_2(i+r) \quad (3)$$

Where i indexes the interrogation windows and r represents pixel displacement. IA_1 and IA_2 are the interrogation window intensities. We use 32-pixel windows, yielding 32 interrogation regions. Given the periodic boundary conditions, the PIV algorithm wraps interrogation windows accordingly.

At this stage, The correlation map is normalized, and vectors with peak correlation values below half the average correlation are set to NaN. Since each window contains at least ten particles, a NaN indicates inadequate particle overlap, as expected in translational motion. Faulty vectors typically exhibit higher correlation values than valid ones (Figure 2).

Sub-pixel estimation is implemented using Gaussian three-point fitting, which is more accurate than centroid-based methods (Lourenco and Krothapalli, 1995; Goodman, 2005; Debella-Gilo and Käab, 2011; Oh et al, 2021). The displacement. The resultant displacement $R(r)$ yields the velocity vector via $v = \frac{dx + correction}{dt}$, where $dt = 2 * 0.01$ from the forward and backward Rk4 stepper.

Last step of this algorithm is outlier detection. The nature of cross correlation does not exclude a possibly of a highly deviant outlier due to poor matching. To account for this, we utilize the statistical approach of median absolute deviation. This is defined

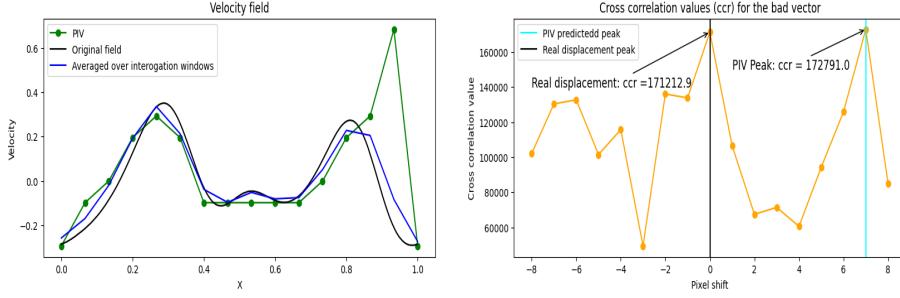


Fig. 2 Bad Vector

as-

$$\begin{aligned} \text{Deviation} &= \text{median}(|X_i - \text{median}(X)|) \\ |X_i - \text{median}(X)| &> k \times \text{Deviation} \end{aligned} \quad (4)$$

Where X is the one dimensional velocity field; i.e. a vector. A loop iterates through each element, flagging values exceeding $k=3$ deviations as outliers, which are then replaced by the average of neighboring vectors. We choose median instead of mean (Westerweel and Scarano, 2005), POD (Higham et al, 2016; Raiola et al, 2015; Raben et al, 2012) or other method of Masullo and Theunissen (2016); Hinsch (1995); Westerweel (1994) as mean-based methods are more sensitive to outliers in small datasets. Furthermore, the threshold is experimentally detected in PIV algorithm(Raffel et al, 2018) finding that threshold is a big challenge here and determining it will be computationally expensive. The correction algorithm may result in inefficient correction in case there is outliers next to each other, and other forms of interpolation may fair better. However, averaging neighbours is much more effective in two dimensional field due to the presence of a higher number of neighbours. Therefore, for consistency we choose to that.

2.2 NN

The training dataset consists of 500,000 pairs of PIV snapshots and their corresponding velocity fields, generated as described in Section 2.1. The validation set comprises 20% of the data, and the entire dataset is shuffled before being split into training and validation sets. Additionally, the PIV snapshots are pre-processed by normalizing their intensities, which involves dividing all snapshots by the maximum intensity across the dataset.

Since FNNs take a vector as input, we concatenate the two PIV snapshots into a single vector of size 2048. For CNNs, we stack the two PIV images. The architectures of both FNN and CNN are shown in Figure 3. Several alternative architectures were tested, but for FNNs, deeper networks consistently produced better results, aligning with the universal approximation theorem (Hornik et al, 1989; Cybenko, 1989). The chosen architecture approaches the RAM limitations of a high-performance GPU. In CNNs, adding max pooling after both convolutional layers resulted in poorer estimates.

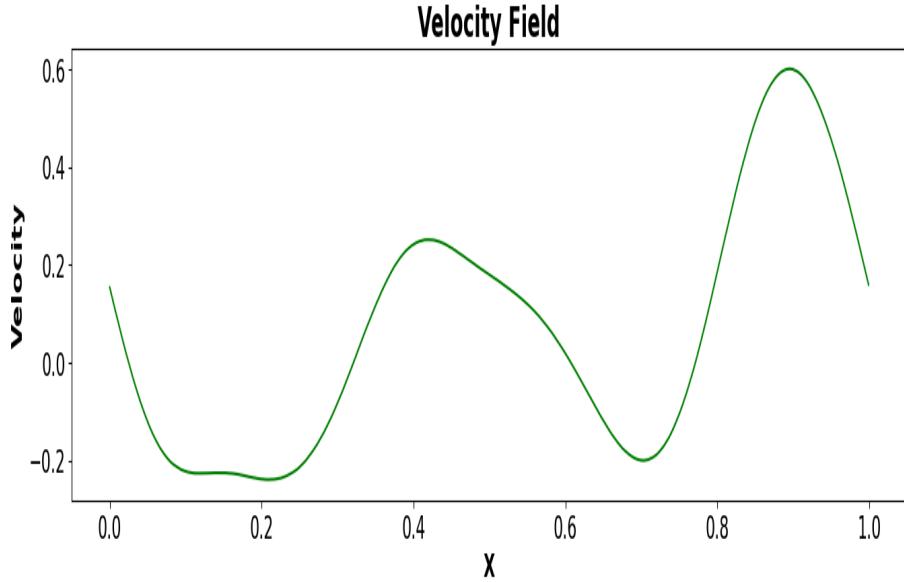


Fig. 3 1D Velocity Field

Various optimizers and activation functions were tested, with RMSprop (See [Tieleman and Hinton, 2012, 2014](#), for details) yielding the best results. RMSprop follows an initial flat phase before proceeding with exponential decay ([Géron, 2022](#)). Since mini-batch optimizers generally perform better on large datasets with redundancy, RMSprop likely outperformed full-batch methods ([Tieleman, 2012](#)).

Two callbacks were implemented during training. First, an early stopping condition halts training if the mean absolute error (MAE) does not improve for 10 epochs. For cases with high noise in the training data, this threshold was increased to 20 epochs to allow the optimizer to move past the initial flat phase. Consequently, any model that failed to improve within 30 epochs was deemed untrainable under the same conditions as the other models. Second, we saved only the model weights corresponding to the lowest MAE, ensuring that modifying the stopping condition did not affect the selection of the best model. An example is shown in Figure 3.

The default batch size of 32 was used. When training on clean data, reducing the batch size to 16 slightly lowered the mean squared error in the training curves. However, smaller batches were more sensitive to noise, leading to early termination by the stopping condition before the optimizer reached its exponential decay stage. This suggests that smaller batches fail to learn effectively in the presence of noise. Increasing the batch size is not expected to improve MAE, and [Géron \(2022\)](#) advises against exceeding the default mini-batch size. Thus, we maintain the default batch size of 32.

The training data was shuffled using a fixed random seed (42) to prevent unintentional correlations between data rows. Training for the clean, noise, gradient, and N&G cases was conducted independently by resetting the kernel before each run, ensuring

a consistent training procedure across all cases. The training curves for all models are provided in the appendix.

3 Result: Constant Fields

Synthetic constant velocity fields represent ideal conditions for PIV cross-correlation algorithms, as they exhibit purely translational flows without any experimental artifacts introduced by laboratory conditions. We evaluate the performance of FNN and CNN on this scenario and compare their results to a custom home-brewed PIV code. The outcomes are presented in Figure 4. As shown, traditional PIV (green bar) achieves a significantly lower MSE, indicating superior performance compared to CNN and FNN for translational flows. This result aligns with expectations given the nature of the setup. However, despite the ideal conditions, questions arise regarding the source of uncertainty in the silent CCR PIV (the leftmost bar).

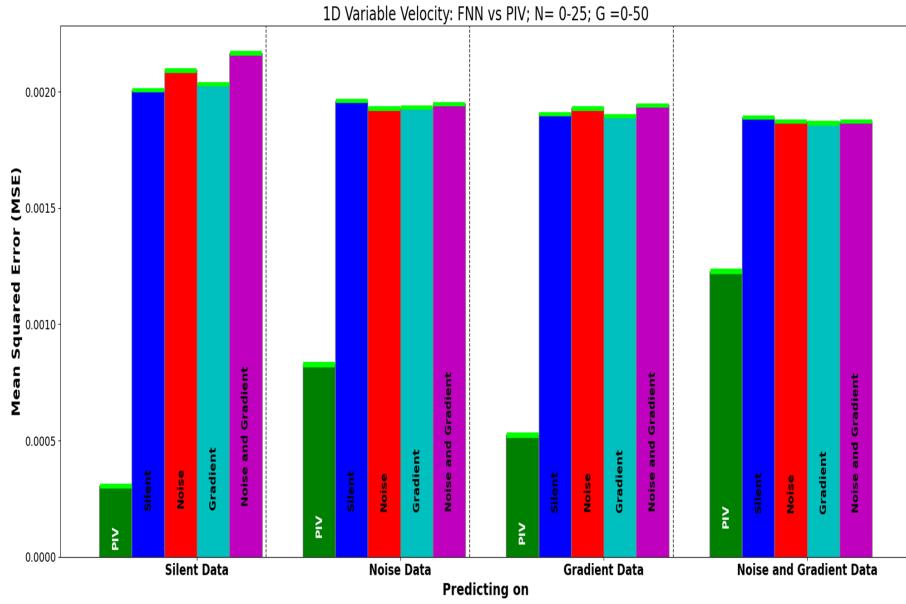


Fig. 4 Caption

We can collective write MSE as follows-

$$\text{MSE} = \sum (\Delta\text{Noise} + \Delta\text{Experimental} + \Delta\text{Systematic}) \quad (5)$$

$$\text{MSE} = \sum_{i=1}^n (((\epsilon_{\text{exp},i})^2 + \epsilon_{\text{noise},i})^2 + (\epsilon_{\text{sys},i})^2) \quad (6)$$

We have chosen ideal conditions for the setup, ensuring that $\epsilon_{\text{experimental}}^2 \rightarrow 0$. Likewise, the leftmost bar represents a case without any added noise, meaning the only

remaining source of uncertainty is the last term. We suspect that this arises from the "pixelation" of the intensity field, which leads to particle confusion and ultimately results in inaccurate vector predictions. A mathematical analysis of PIV noise statistics by [George and Stanislas \(2021\)](#); [Lourenco \(2000\)](#); [Foucaut et al \(2004\)](#) highlights that pixelation prior to integration introduces errors specific to PIV. Unlike other methods, such as HWA, where pixelation effects can be treated as an additive term, PIV is directly affected by this phenomenon. [George and Stanislas \(2021\)](#) derived [Equation 7](#), which we reference to highlight the contribution of the pixelization term

$$o_i(\mathbf{y}, t) = \frac{1}{N} \int_{\mathcal{S}} [v_i(\mathbf{a}, t) + P_i(\mathbf{a}, t)] w(\mathbf{y}, \mathbf{X}[\mathbf{a}, t] - \mathbf{y}) g(\mathbf{a}) d\mathbf{a} \quad (7)$$

Where $P_i(\mathbf{a}, t), v_i(\mathbf{a}, t)$ correspond to the pixelization term, individual lagrangian velocity. Therefore, as the $P_i(\mathbf{a}, t) \rightarrow 0$ then $\epsilon_{\text{systemtic}}^2 \rightarrow 0$ then consequently $MSE \rightarrow 0$. Investigations pertaining to the uncertainty of ccr algorithms show that at 1/2 a pixel the MSE appears to be greatest along with the sense of quantization. This is demonstrated in [Figure 5](#). The results are consistent with experimental investigation and Monte Carlo simulations ([Dabiri, 2009](#); [Adrian and Westerweel, 2011](#); [Huang et al, 1997](#)).

Moving along to the other ccr PIV bars, it is evident that at low intensity, noise is the major contributing factor to the MSE and the gradient follows second. An example comparing the prediction is shown in [Figure 6](#). It is clearly evident that CCR has the most accurate prediction for the Silent case with outliers only caused by pixelation. Therefore, collectively the leftmost bar demonstrates the MSE caused by outlier which are a result of pixelation. If an ideal outlier detection algorithm is used then

$$\lim_{\text{outliers} \rightarrow 0} MSE \rightarrow 0 \quad (8)$$

The noise and gradient increase the number of outliers. It is also worth noting that the neural networks are roughly 10% percent off on the accuracy of the velocity field. Further, neural networks appears to learn the constant nature of the velocity field and therefore do not produce any outliers even in the case of noise and gradient. It is likely due to this reason that their accuracy suffers much less in the high noise cases. Therefore, we conclude that in cases where the nature of the velocity field is known but experimental conditions introduce significant disturbance then it may be better suited for a CNN.

4 Result: Random Velocity Fields

Similar to the constant velocity field and literature studies, the variable velocity field also shows quantization in error. The error increase as the velocity increases. Similarly, the error is the lowest at integer pixel displacements. A plot similar to Figure 5 is on the pc. Consesus on FNN vs CNN is: CNN as much better pretty close to PIV for low intensity noise. FNN are just not upto the task.

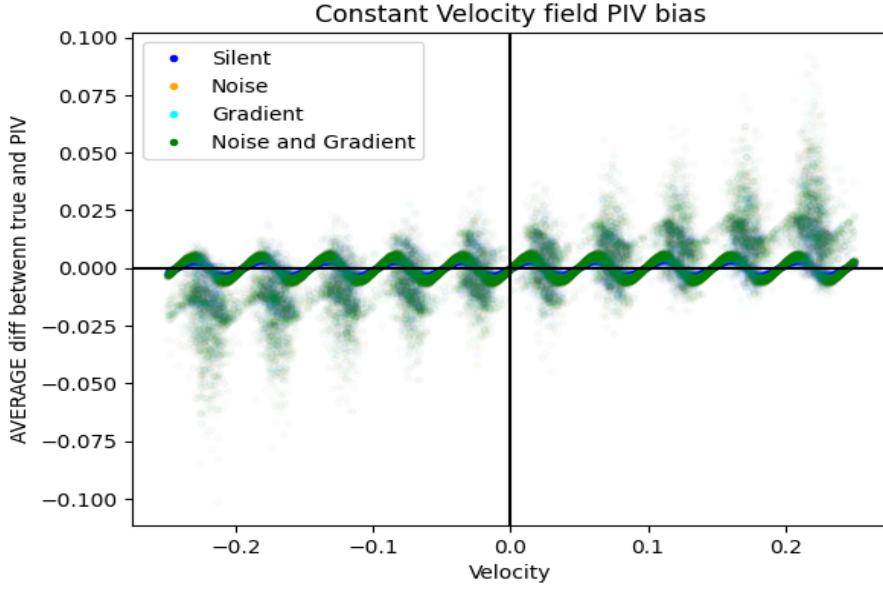


Fig. 5 Change x axis to displacement

4.1 Function of Noise intensity

Regarding Figure 7 Gaussian white noise of higher intensity in the background leads to more outliers. Implementing the outlier detector will show that PIV is better than NN. FNN and CNN have a very challenging time when predicting on data with high intensity noise as evident by looking at the velocity field rather than ensemble MSE.

4.2 Function of Training data

Figure 8 is a plot showing the difference in prediction when the CNN is trained on low and high noise. Evidently it is better to train the CNN on clean data when the amount of noise in the test sample is low. Conversely, data with high intensity noise and gradient is better predicted on by CNN trained copious noise data. To conclude, in case of a polluted sample, a NN trained on the polluted sample is more accurate and vice versa. However, a NN trained on noisy sample will predict poorly on clean data as it learns to expect noise.

4.3 Function of σ

σ controls how abruptly the velocity field changes. This is demonstrated in Figure 9.

In Figure 10 we can see that PIV error is much higher than CNN at $\sigma = 0.025$. This is likely because there is a change in the velocity within an interrogation window, and therefore all the particles in a window do not follow the translational flow in the same direction i.e. a velocity gradient within the window.

It can also be noticed that at high σ there is a small wiggle that appears like perturbation. This is because the width of the sigma is bigger than the space leading

Variable Velocity Field: FNN Predicting on Corresponding Data

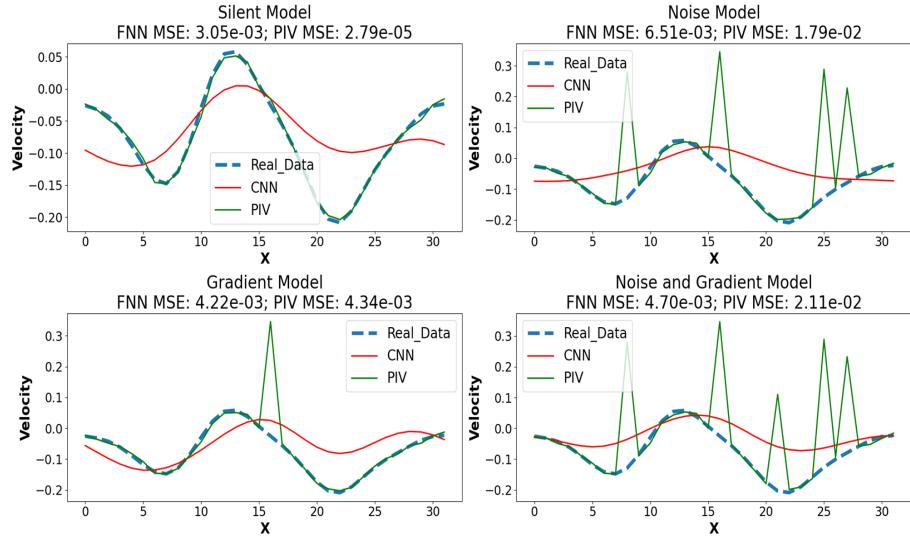


Fig. 6 Caption

to confusion in the closest point given periodic boundary i.e. it is taking distance from both sides. This is an error which needs to be corrected. However, this puts forth the idea that small scale perturbations in the velocity field hardly deter the CNN's ability to predict on an otherwise smooth field as compared to Cross-correlation which suffers noticeably.

The graph shows that training it on $\sigma = 0.100$ does not create any bias, improve or inclination for the CNN to predict better on the field with the same statistic. Instead, it predicts the best on $\sigma = 0.200$; the straightest, smoothest ,and overall easiest velocity field to determine. Therefore, CNN (and likely all NN) are not bias towards the statistic that they were trained on.

5 Discussion

6 Conclusion and Summary

References

Adatrao S, Bertone M, Sciacchitano A (2021) Multi- Δt approach for peak-locking error correction and uncertainty quantification in PIV. Measurement Science and Technology 32(5):054003. <https://doi.org/10.1088/1361-6501/abdcde>

Adrian RJ (2005) Twenty years of particle image velocimetry. Experiments in Fluids 39(2):159–169. <https://doi.org/10.1007/s00348-005-0991-7>

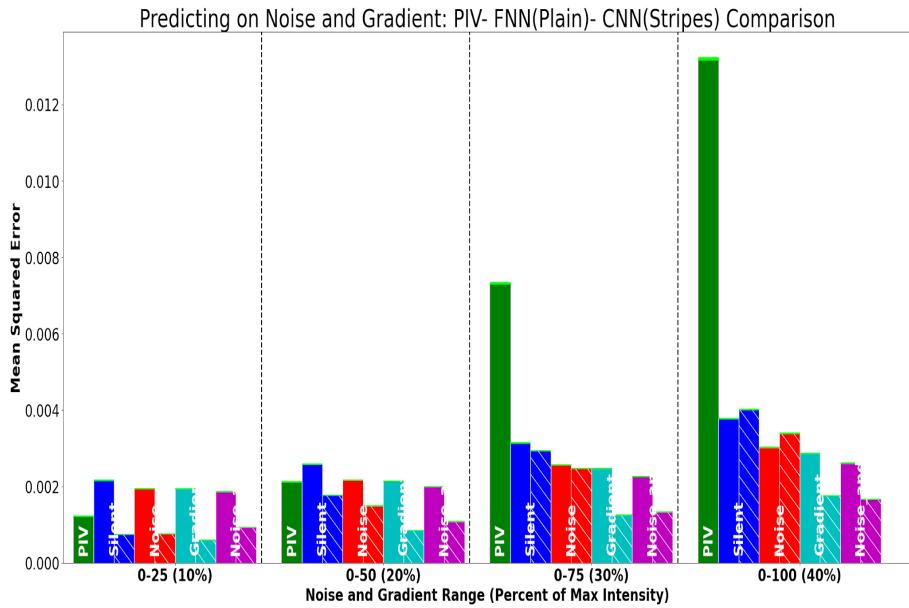


Fig. 7

Adrian RJ, Westerweel J (2011) Particle image velocimetry. 30, Cambridge university press

Augier P, Campagne A, Valran T, et al (2017) Measuring mixing efficiency in experiments of strongly stratified turbulence. In: AGU Fall Meeting Abstracts, pp NG21A-0127

Bastiaans RJM (1993) Cross-correlation PIV: Theory, implementation and accuracy. Technische Universiteit Eindhoven

Beresh SJ (2021) Time-resolved particle image velocimetry. Measurement Science and Technology 32(10):102003. <https://doi.org/10.1088/1361-6501/ac08c5>

Butcher JC (1996) A history of runge-kutta methods. Applied numerical mathematics 20(3):247–260

Cai S, Liang J, Zhou S, et al (2019) Deep-piv: A new framework of piv using deep learning techniques. In: Proceedings of the 13th International Symposium on Particle Image Velocimetry—ISPIV

Calicchia MA, Mittal R, Seo JH, et al (2023) Reconstructing the pressure field around swimming fish using a physics-informed neural network. Journal of Experimental Biology 226(8):jeb244983

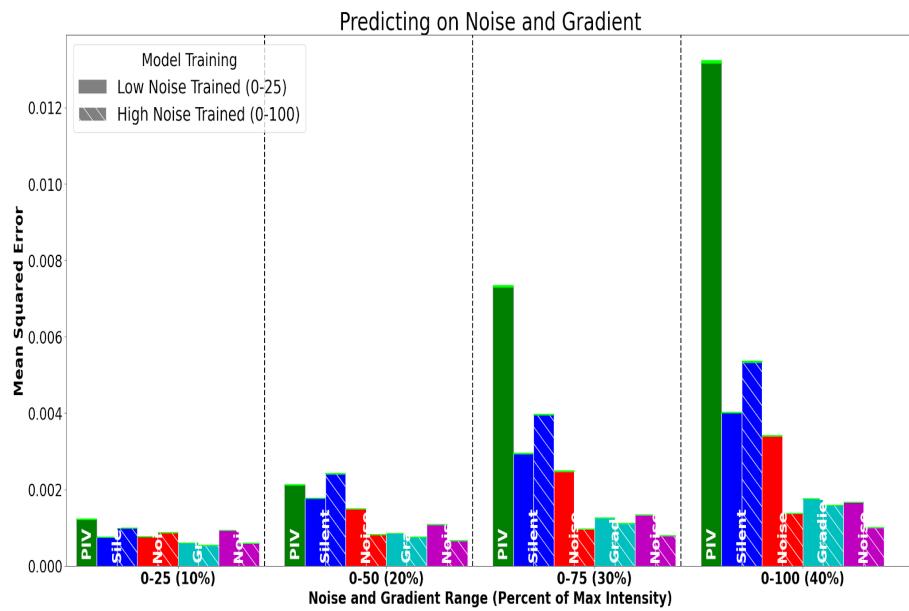


Fig. 8

Cameron SM (2022) Theoretical description of PIV measurement errors. *Acta Geophysica* 70(5):2379–2387. <https://doi.org/10.1007/s11600-022-00901-9>

Carosone F, Cenedese A, Querzoli G (1995) Recognition of partially overlapped particle images using the Kohonen neural network. *Experiments in Fluids* 19(4):225–232. <https://doi.org/10.1007/BF00196470>

Cybenko G (1989) Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems* 2(4):303–314

Dabiri D (2009) Digital particle image thermometry/velocimetry: a review. *Experiments in Fluids* 46:191–241

Debella-Gilo M, Käab A (2011) Sub-pixel precision image matching for measuring surface displacements on mass movements using normalized cross-correlation. *Remote Sensing of Environment* 115(1):130–142. <https://doi.org/10.1016/j.rse.2010.08.012>

Discetti S, Liu Y (2023) Machine learning for flow field measurements: a perspective. *Measurement Science and Technology* 34(2):021001. <https://doi.org/10.1088/1361-6501/ac9991>

Discetti S, Sciacchitano A (2023) Special Issue on Machine learning and data assimilation techniques for fluid flow measurements. *Measurement Science and Technology* 34(9):090201. <https://doi.org/10.1088/1361-6501/acce53>

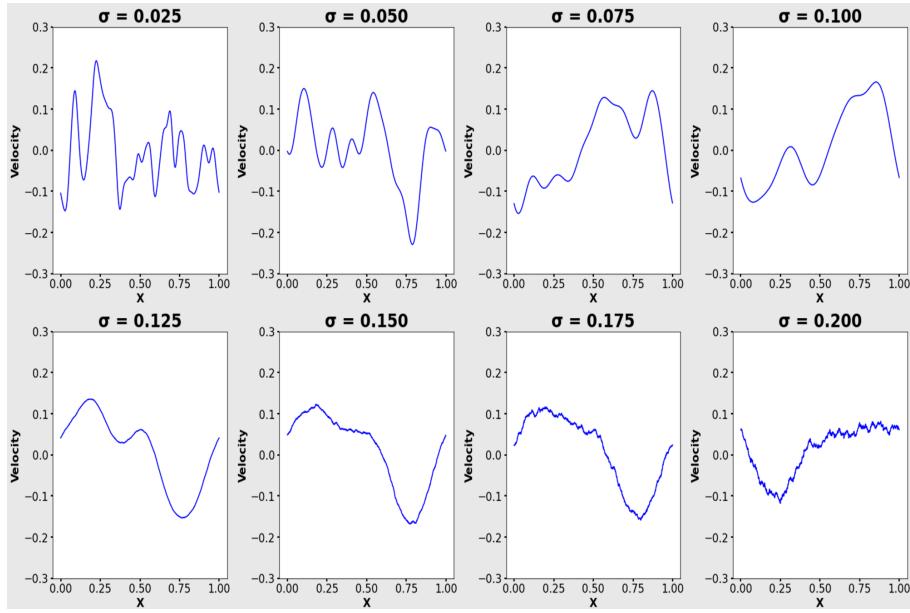


Fig. 9

Eivazi H, Wang Y, Vinuesa R (2024) Physics-informed deep-learning applications to experimental fluid mechanics. Measurement Science and Technology 35(7):075303. <https://doi.org/10.1088/1361-6501/ad3fd3>, arXiv:2203.15402 [physics.flu-dyn]

Etminan A, Muzychka YS, Pope K, et al (2022) Flow visualization: state-of-the-art development of micro-particle image velocimetry. Measurement Science and Technology 33(9):092002. <https://doi.org/10.1088/1361-6501/ac75b0>

Foucaut JM, Carlier J, Stanislas M (2004) PIV optimization for the study of turbulent flow using spectral analysis. Measurement Science and Technology 15(6):1046–1058. <https://doi.org/10.1088/0957-0233/15/6/003>

G. Yazdi S, Mercier D, Bernard R, et al (2020) Particle image velocimetry measurements of the flow-diverting effects of a new generation of the eclips implant for the treatment of intracranial bifurcation aneurysms. Applied Sciences 10(23):8639

Gauss KF (1809) Theoria motus corporvm coelestivm in sectionibvs conicis solem ambientivm. FA Perthes

George WK, Stanislas M (2021) On the noise in statistics of PIV measurements. Experiments in Fluids 62(9):188. <https://doi.org/10.1007/s00348-021-03163-1>, arXiv:2010.10768 [physics.flu-dyn]

Géron A (2022) Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. ” O'Reilly Media, Inc.”

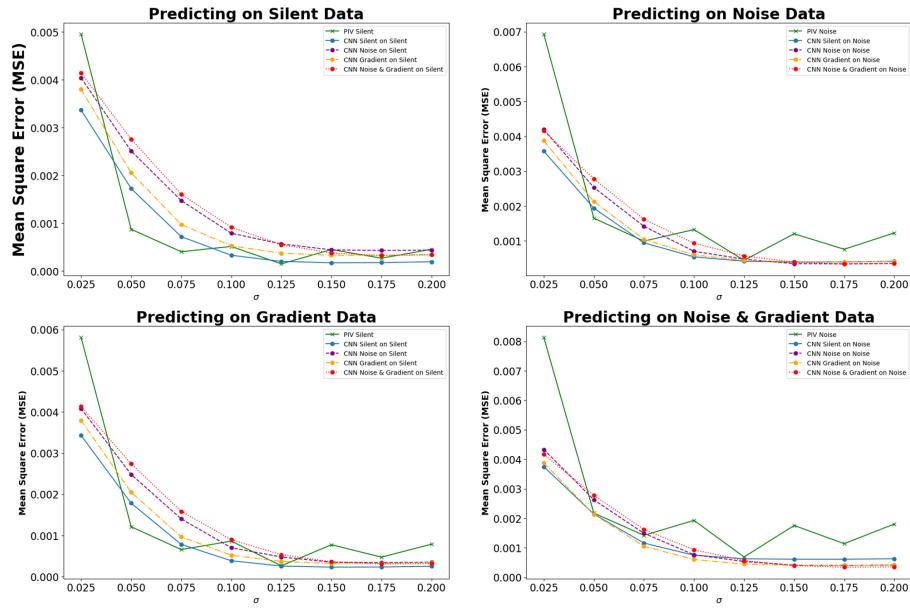


Fig. 10

Goodman JW (2005) Introduction to Fourier optics

Grant I, Pan X (1993) Neural network method applied to particle image velocimetry. In: Cha SS, Trolinger JD (eds) Optical Diagnostics in Fluid and Thermal Flow, pp 437–447, <https://doi.org/10.1111/12.163728>

Grant I, Pan X (1997) The use of neural techniques in PIV and PTV. Measurement Science and Technology 8(12):1399–1405. <https://doi.org/10.1088/0957-0233/8/12/004>

Hassan N, Zawawi M, Al Bakri AM, et al (2020) A review on applications of particle image velocimetry. In: IOP Conference Series: Materials Science and Engineering, IOP Publishing, p 012149

Hassan YA, Philip OG (1997) A new artificial neural network tracking technique for particle image velocimetry. Experiments in Fluids 23(2):145–154. <https://doi.org/10.1007/s003480050096>

Higham JE, Brevis W, Keylock CJ (2016) A rapid non-iterative proper orthogonal decomposition based outlier detection and correction for PIV data. Measurement Science and Technology 27(12):125303. <https://doi.org/10.1088/0957-0233/27/12/125303>

Hinsch KD (1995) Three-dimensional particle velocimetry. Measurement Science and Technology 6(6):742–753. <https://doi.org/10.1088/0957-0233/6/6/012>

Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. *Neural networks* 2(5):359–366

Huang H, Dabiri D, Gharib M (1997) On errors of digital particle image velocimetry. *Measurement Science and Technology* 8(12):1427–1440. <https://doi.org/10.1088/0957-0233/8/12/007>

Johnson BA (2020) Exploring the role of bed fluidization on ripple formation in highly turbulent flows. In: AGU Fall Meeting Abstracts, pp OS035–03

Johnson RW (2016) Handbook of fluid dynamics. CRC press

Kawahashi M, Hirahara H (2000) Velocity and density field measurements by digital speckle method. *Optics Laser Technology* 32(7):575–582. [https://doi.org/10.1016/S0030-3992\(00\)00089-X](https://doi.org/10.1016/S0030-3992(00)00089-X)

Keane RD, Adrian RJ (1992) Theory of cross-correlation analysis of PIV images. *Flow, Turbulence and Combustion* 49(3):191–215. <https://doi.org/10.1007/BF00384623>

Kutta W (1901) Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. Teubner

Lagemann C, Lagemann K, Mukherjee S, et al (2022) Generalization of deep recurrent optical flow estimation for particle-image velocimetry data. *Measurement Science and Technology* 33(9):094003. <https://doi.org/10.1088/1361-6501/ac73db>

Lee Y, Yang H, Yin Z (2017) PIV-DCNN: cascaded deep convolutional neural networks for particle image velocimetry. *Experiments in Fluids* 58(12):171. <https://doi.org/10.1007/s00348-017-2456-1>

Lindken R, Westerweel J, Wieneke B (2006) 3d micro-scale velocimetry methods: A comparison between 3d- μ ptv, stereoscopic μ piv and tomographic μ piv. In: 13th Int. Symp. on Applications of Laser Techniques to Fluid Mechanics, June, pp 26–29

Loureiro L, Krothapalli A (1995) On the accuracy of velocity and vorticity measurements with PIV. *Experiments in Fluids* 18(6):421–428. <https://doi.org/10.1007/BF00208464>

Loureiro LM (2000) True resolution piv: a mesh-free second order accurate algorithm. In: International Conference of Lasers to Fluid Mechanics

Luzzatto-Fegiz P (2022) Simultaneous PIV and LIF measurements in stratified flows using pulsed lasers. arXiv e-prints arXiv:2207.05455. <https://doi.org/10.48550/arXiv.2207.05455>, arXiv:2207.05455 [physics.flu-dyn]

Majewski W, Kaspera W, Ples M, et al (2021) Super-resolution study of aneurism based on AI PIV - preliminary report. In: APS Division of Fluid Dynamics Meeting Abstracts, APS Meeting Abstracts, p H14.010

- Masullo A, Theunissen R (2016) Adaptive vector validation in image velocimetry to minimise the influence of outlier clusters. *Experiments in Fluids* 57:33. <https://doi.org/10.1007/s00348-015-2110-8>
- McCutchan A, Johnson BA (2020) The Investigation of Ice Melting Rates in Homogeneous Isotropic Turbulence. In: AGU Fall Meeting Abstracts, pp NG012–01
- Mendez MA, Dominique J, Fiore M, et al (2022) Challenges and Opportunities for Machine Learning in Fluid Mechanics. arXiv e-prints arXiv:2202.12577. <https://doi.org/10.48550/arXiv.2202.12577>, arXiv:2202.12577 [physics.flu-dyn]
- Michaelis D, Neal DR, Wieneke B (2016) Peak-locking reduction for particle image velocimetry. *Measurement Science and Technology* 27(10):104005. <https://doi.org/10.1088/0957-0233/27/10/104005>
- Nieuwstadt FTM, Westerweel J, Boersma BJ (2016) Turbulence: Introduction to Theory and Applications of Turbulent Flows. Springer International Publishing, <https://doi.org/10.1007/978-3-319-31599-7>
- Nogueira J, Lecuona A, Nauri S, et al (2011) Quantitative evaluation of PIV peak locking through a multiple Δt strategy: relevance of the rms component. *Experiments in Fluids* 51(3):785–793. <https://doi.org/10.1007/s00348-011-1094-2>
- Nogueira J, Legrand M, Jiménez R, et al (2021) Peak-locking full characterization: PIV error assessment and velocity ensemble measurement correction. *Measurement Science and Technology* 32(11):114005. <https://doi.org/10.1088/1361-6501/ac092a>
- Oh JS, Lee H, Hwang W (2021) Motion blur treatment utilizing deep learning for time-resolved particle image velocimetry. *Experiments in Fluids* 62(11):234. <https://doi.org/10.1007/s00348-021-03330-4>
- Penland C (1996) A stochastic model of IndoPacific sea surface temperature anomalies. *Physica D Nonlinear Phenomena* 98(2):534–558. [https://doi.org/10.1016/0167-2789\(96\)00124-8](https://doi.org/10.1016/0167-2789(96)00124-8)
- Prandtl L (1905) Über flüssigkeitsbewegung bei sehr kleiner reibung. Verhandl 3rd Int Math Kongr Heidelberg (1904), Leipzig
- Prandtl L (1936) Entstehung von wirbeln bei wasserströmungen:–1. Entstehung von Wirbeln und Künstliche Beeinflussung der Wirbelbildung Institut für Wissenschaftlichen Film (DVD)–Historische Filmaufnahmen
- Prasad AK (2000) Stereoscopic particle image velocimetry. *Experiments in Fluids* 29(2):103–116. <https://doi.org/10.1007/s003480000143>
- Rabault J, Kolaas J, Jensen A (2017) Performing particle image velocimetry using artificial neural networks: a proof-of-concept. *Measurement Science and Technology*

28(12):125301. <https://doi.org/10.1088/1361-6501/aa8b87>

Raben SG, Charonko JJ, Vlachos PP (2012) Adaptive gappy proper orthogonal decomposition for particle image velocimetry data reconstruction. *Measurement Science and Technology* 23(2):025303. <https://doi.org/10.1088/0957-0233/23/2/025303>

Raffel M, Willert CE, Scarano F, et al (2018) Particle image velocimetry: a practical guide. Springer

Raiola M, Discetti S, Ianiro A (2015) On PIV random error minimization with optimal POD-based low-order reconstruction. *Experiments in Fluids* 56:75. <https://doi.org/10.1007/s00348-015-1940-8>

Reynolds O (1883) Xxix. an experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous, and of the law of resistance in parallel channels. *Philosophical Transactions of the Royal Society of London* (174):935–982

Rohacs D, Yasar O, Kale U, et al (2023) Past and current components-based detailing of particle image velocimetry: A comprehensive review. *Heliyon* 9(3)

Runge C (1895) Über die numerische auflösung von differentialgleichungen. *Mathematische Annalen* 46(2):167–178

Scarano F (2013) Tomographic PIV: principles and practice. *Measurement Science and Technology* 24(1):012001. <https://doi.org/10.1088/0957-0233/24/1/012001>

Scharnowski S, Kähler CJ (2020) Particle image velocimetry-classical operating rules from today's perspective. *Optics and Lasers in Engineering* 135:106185

Sciacchitano A, Wieneke B, Scarano F (2013) PIV uncertainty quantification by image matching. *Measurement Science and Technology* 24(4):045302. <https://doi.org/10.1088/0957-0233/24/4/045302>

Seelig T, Harlander U, Gellert M (2019) Experimental investigation of stratorotational instability using a thermally stratified system: instability, waves and associated momentum flux. In: AGU Fall Meeting Abstracts, pp NG43A–0902

Stahl S (2006) The evolution of the normal distribution. *Mathematics magazine* 79(2):96–113

Stamhuis E, Thielicke W (2014) Pivlab—towards user-friendly, affordable and accurate digital particle image velocimetry in matlab. *Journal of open research software* 2(1):30

Tieleman T (2012) Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning 4(2):26

- Tieleman T, Hinton G (2012) Rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning. COURSERA Neural Networks Mach Learn 17
- Tieleman T, Hinton G (2014) Rmsprop gradient optimization. URL http://www.cs.toronto.edu/tijmen/csc321/slides/lecture_slides_lec6.pdf
- Viquerat J, Meliga P, Larcher A, et al (2022) A review on deep reinforcement learning for fluid mechanics: An update. Physics of Fluids 34(11):111301. <https://doi.org/10.1063/5.0128446>, arXiv:2107.12206 [physics.flu-dyn]
- Westerweel J (1994) Efficient detection of spurious vectors in particle image velocimetry data. Experiments in Fluids 16(3-4):236–247. <https://doi.org/10.1007/BF00206543>
- Westerweel J, Scarano F (2005) Universal outlier detection for PIV data. Experiments in Fluids 39(6):1096–1100. <https://doi.org/10.1007/s00348-005-0016-6>
- Westerweel J, Elsinga GE, Adrian RJ (2013) Particle Image Velocimetry for Complex and Turbulent Flows. Annual Review of Fluid Mechanics 45(1):409–436. <https://doi.org/10.1146/annurev-fluid-120710-101204>
- Wheeler J, Helfrich K, Anderson E, et al (2013) Upward swimming of competent oyster larvae *Crassostrea virginica* persists in highly turbulent flow as detected by PIV flow subtraction. Marine Ecology Progress Series 488:171–185. <https://doi.org/10.3354/meps10382>
- Willert C, Kompenhans J (2010) PIV Analysis of Ludwig Prandtl's Historic Flow Visualization Films. arXiv e-prints arXiv:1010.3149. <https://doi.org/10.48550/arXiv.1010.3149>, arXiv:1010.3149 [physics.flu-dyn]
- Yu C, Bi X, Fan Y (2023) Deep learning for fluid velocity field estimation: A review. Ocean Engineering 271:113693
- Yu C, Fan Y, Bi X, et al (2023) Deep dual recurrence optical flow learning for time-resolved particle image velocimetry. Physics of Fluids 35(4):045104. <https://doi.org/10.1063/5.0142604>

Appendices

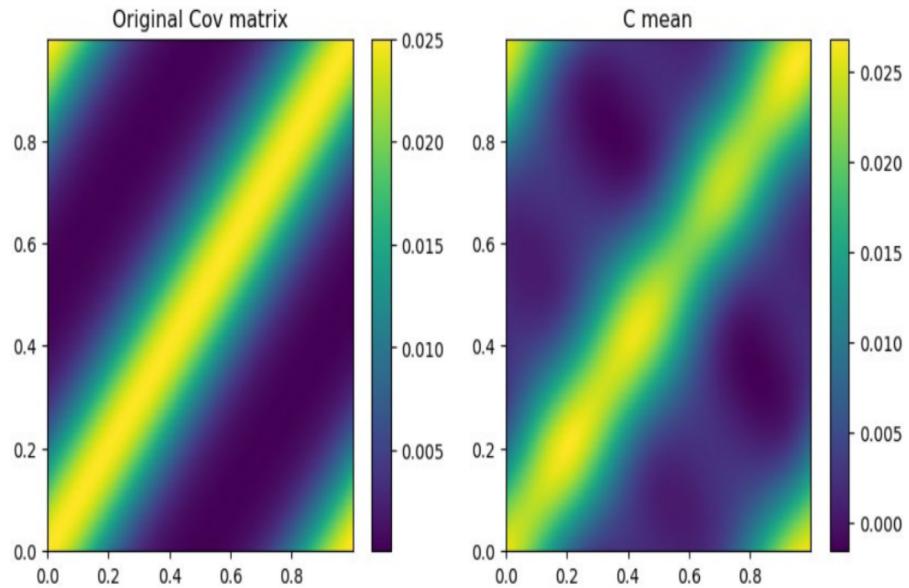


Fig. 11 The original and reconstructed covariance matrix. Setting negative eigenvalues to 0 and only considering real eigenvectors has nominal impact

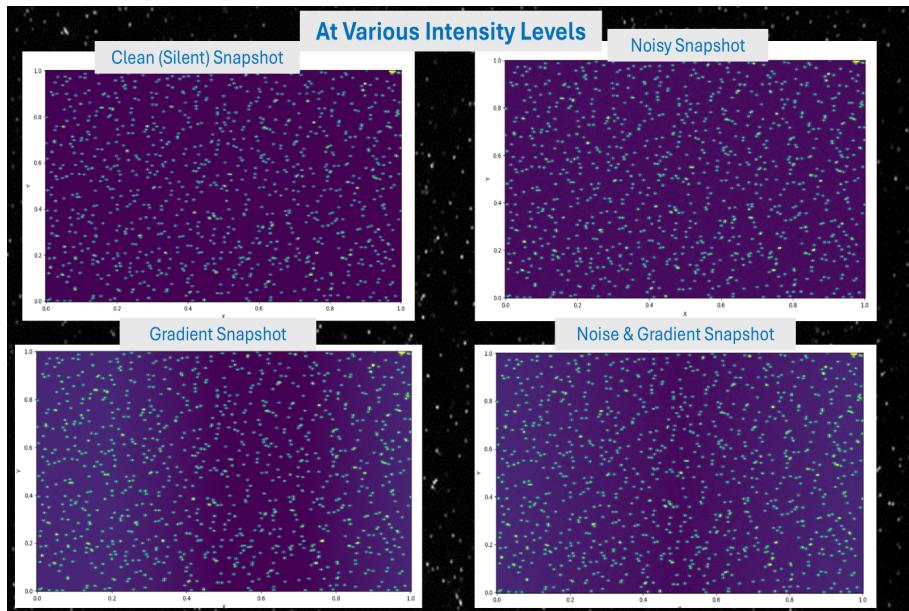


Fig. 12

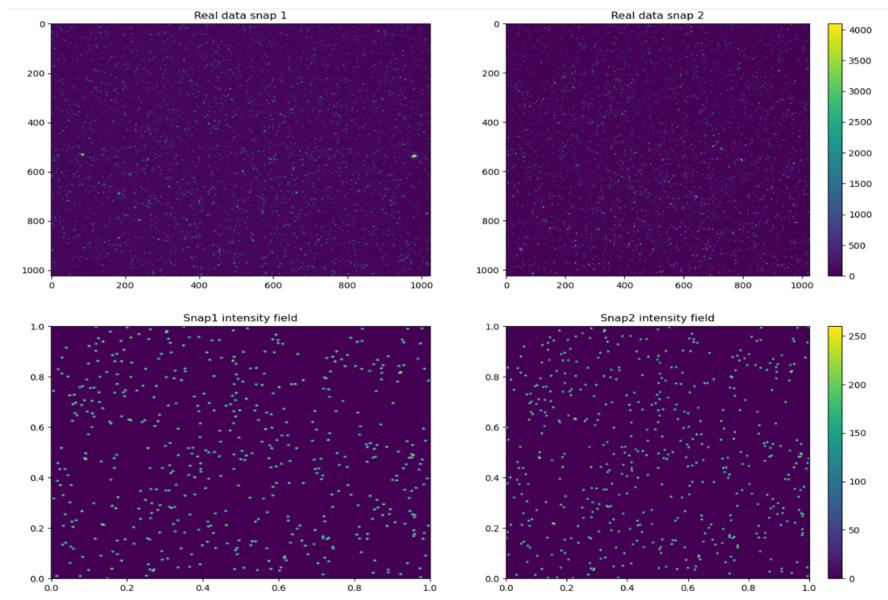


Fig. 13

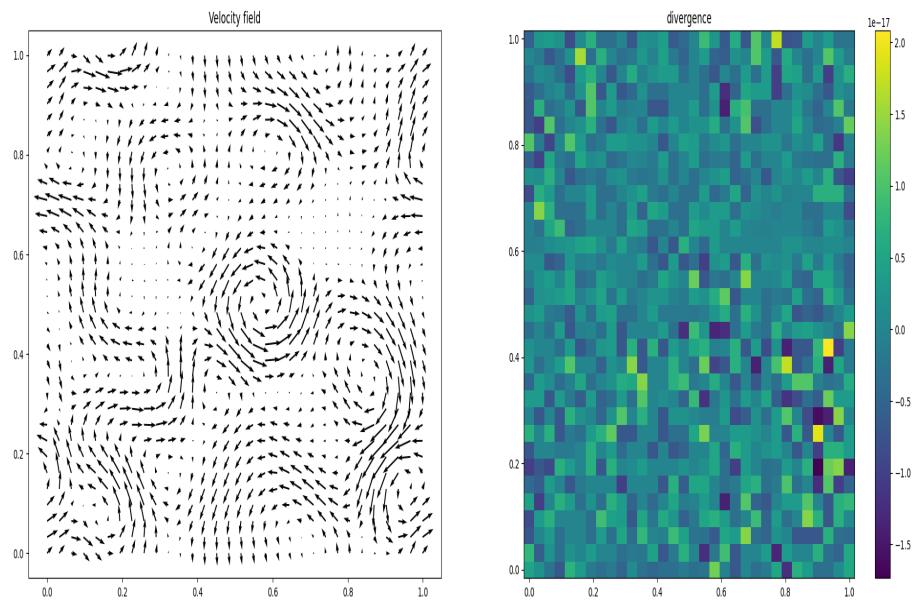


Fig. 14 2DVelocity