16th International Symposium on Particle Image Velocimetry – ISPIV
2025 June 26–28, 2025, Tokyo, Japan

Robust Flow Field Reconstruction Using PINN for 3D Lagrangian Particle Tracking

H. Shin[1*], A. Schröder[1,2]

[1] German Aerospace Center (DLR), Inst. of Aerodynamics and Flow Technology, Göttingen, Germany
[2] Brandenburg Univ. of Techn., Dep. of Image Based Measurement Techniques, Cottbus, Germany
*Shin.Hyungmin@dlr.de

Abstract

Physics-Informed Neural Networks (PINNs) effectively reconstruct fluid flow fields from sparse Lagrangian Particle Tracking (LPT) data by embedding physical laws directly into neural network training. This study investigates the influence of experimental parameters—such as particle density—on PINN reconstruction performance using synthetic (DNS-based HIT and turbulent channel flows) and experimental turbulent boundary layer (TBL) datasets. Results demonstrate PINNs' robustness across varying seeding densities, with notably superior performance in TBL cases compared to HIT cases. Optimization techniques, particularly the SOAP optimizer, significantly enhance convergence speed and accuracy, highlighting PINNs' potential for reliable fluid flow reconstruction from limited experimental data.

1 Introduction

Lagrangian Particle Tracking (LPT) is a well-established experimental technique used to analyze the trajectories of individual particles, allowing for a detailed quantitative visualization of fluid flow behavior. Unlike other volumetric measurement techniques, such as Tomographic Particle Image Velocimetry (Tomo-PIV), which average data over spatial grids, LPT tracks individual particles as they move through the flow field. This approach provides direct measurements of trajectories, velocities and accelerations with high spatial resolution (Schröder and Schanz 2023). A significant advancement in LPT is the development of the Shake-The-Box (STB) algorithm (Schanz et al. 2016), which enables time-resolved 3D LPT for densely seeded flows, making it possible to analyze complex fluid dynamics, such as turbulence, in greater detail. However, several challenges remain, particularly in calculating spatial gradients and reconstructing pressure fields—both of which are critical areas of research in fluid dynamics (Sciacchitano et al. 2025).

Reconstructing the full flow field onto a structured grid, including partial velocity gradients and the pressure field from LPT data, is commonly referred to as Cartesian Grid Reduction (CGR). This reconstruction can be achieved using various data assimilation (DA) techniques. A straightforward approach is the ensemble-averaging method, which averages velocity vectors from LPT data within small spatial bins (Agüera et al. 2016; Kasagi and Nishino 1991). Although this method is simple and easy to implement, it does not inherently enforce physical constraints. To incorporate physical laws into the CGR process, for example the DA method FlowFit introduces the continuity and momentum equations along with particle acceleration data to determine the optimal coefficients of a 3D B-spline basis that represents the flow field (Godbersen et al. 2024). This method has been further improved by incorporating divergence-freeness as a hard constraint, restricting the optimizer to search within a divergence-free subspace of the velocity field. Similarly, Vortex-In-Cell (VIC) algorithms (Schneiders et al. 2014; Schneiders and Scarano 2016; Jeon et al. 2022)—such as VIC+ and VIC#—enforce the vorticity dynamic equation in conjunction with measured velocity and acceleration data.

With the advent of Physics-Informed Neural Networks (PINNs) in the scientific community (Raissi et al. 2019), multiple successful applications of PINNs to LPT data have been reported. Wang et al. demonstrated PINN-based reconstruction of simulated LPT data from direct numerical simulation

(DNS) of channel flow (H. Wang et al. 2022). Clark di Leoni et al compared PINN reconstruction performance with the conventional DA method, Constrained Cost Minimization (CCM), in reconstructing synthetic DNS channel flow and experimental turbulent shear flow (Clark Di Leoni et al. 2023). Cai et al. showed that PINNs can achieve super-resolution velocity field reconstruction in both space and time from sparse measurements in tomographic PTV experiments of jet flow in water (Cai et al. 2024). Diego Toscano et al. introduced a new method, AIVT, based on the Kolmogorov-Arnold network, for temperature inference in Rayleigh-Bénard Convection (RBC) cells (Toscano et al. 2024). Steinfurth et al. compared the PINN approach to the binning method and Vortex-In-Cell with Time-Space Assimilation (VIC-TSA) under different data sparsity conditions in pulsed jets (Steinfurth et al. 2024).

PINNs have proven to be highly effective in the DA field for both numerical and experimental datasets. However, deeper insight into the relationship between LPT experimental parameters and PINN reconstruction performance—particularly in preserving fluid flow gradients—remains an open research question. In this study, we evaluate the influence of key LPT parameters, such as particle concentration rate in the volume and camera repetition rate, on the reconstruction performance of PINNs for both numerical and experimental test cases.

We present several PINN methodologies that have proven to be among the most effective for reconstructing flow structures from sparse particle clouds. Section 2 provides a fundamental overview of PINNs, while Section 3 describes the test cases used to evaluate PINN-based DA, ranging from synthetic LPT-simulated datasets to actual experimental datasets obtained via the STB method. In Section 4, we present the results of PINN-based flow field reconstructions across various turbulent fluid flow scenarios.

2 Physics informed neural network

A Physics-Informed Neural Network is a deep neural network-based global function approximator that embeds governing physical laws by leveraging automatic differentiation of the output components with respect to the input values. The general architecture of a PINN consists of a neural network with trainable parameters θ, which receives input coordinates and returns the predicted outputs. The loss function comprises terms based on the governing equations and measured data, as illustrated in Figure 1.

Given velocity data at specific particle locations extracted from fitted LPT trajectories, the PINN interpolates the velocity field by minimizing the discrepancy between predicted and experimental velocities. The data loss term is defined as:

$$L_d = \frac{1}{N_d}\sum_{j=1}^{N_d}\left|u_j - \hat{u}_j\right|^2 \tag{1}$$

where u_i, $\hat{u}_i$, and N_d denote the measured velocity, predicted velocity, and number of input vectors, respectively. While fitting the velocity field, the PINN also enforces physical constraints through the governing equations for incompressible flow: the continuity equation and the incompressible Navier-Stokes equation. These constraints are included via the following loss terms:

$$L_i = \frac{1}{N_p}\sum_{j=1}^{N_p}\left|\nabla \cdot u_j\right|^2 \tag{2}$$

$$L_m = \frac{1}{N_p}\sum_{j=1}^{N_p}\left|\frac{\partial u_j}{\partial \mathrm{t}} + u_j \cdot \nabla u_j + \frac{\nabla \mathrm{p}}{\rho} - \nu \nabla^2 u_j\right|^2 \tag{3}$$

Here t, p, ρ, ν, and N_p represent time, pressure, fluid density, kinematic viscosity, and the number of residual points, respectively. Since the magnitude of each loss term can vary depending on the flow characteristics, balancing them is essential to ensure stable convergence. The total loss function is defined as:

$$\mathrm{L} = \lambda_d L_d + \lambda_i L_i + \lambda_m L_m \tag{4}$$

where λ_d, λ_i, and λ_m are weighting factors that control the influence of each term.

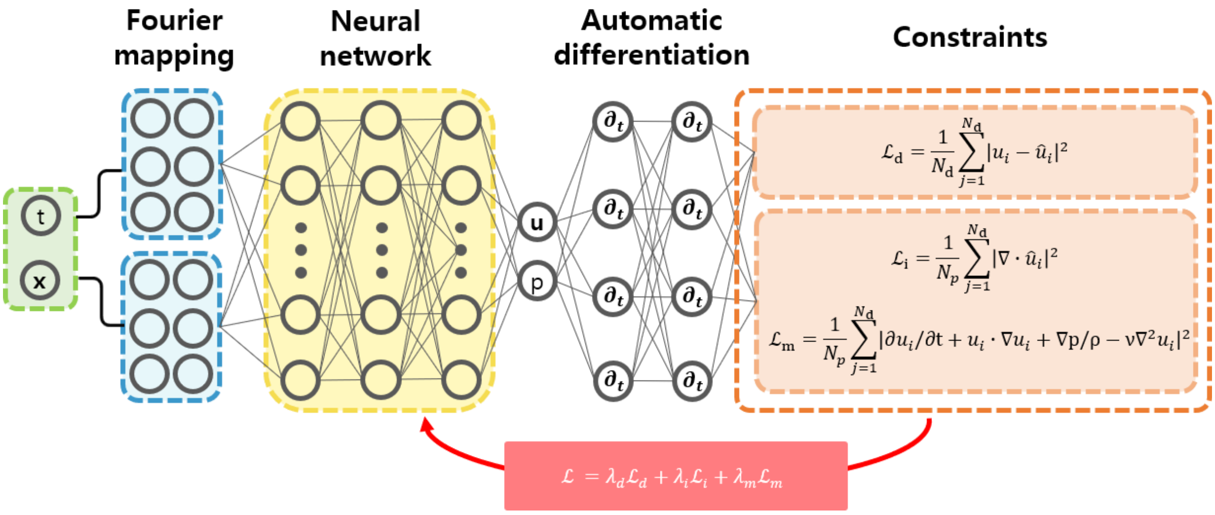

Fourier mapping
Neural network
Automatic differentiation
Constraints
t
x
u
p
∂_t
$\mathcal{L}_\mathrm{d} = \frac{1}{N_\mathrm{d}}\sum_{j=1}^{N_\mathrm{d}} |u_i - \hat{u}_i|^2$
$\mathcal{L}_\mathrm{i} = \frac{1}{N_p}\sum_{j=1}^{N_\mathrm{d}} |\nabla \cdot \hat{u}_i|^2$
$\mathcal{L}_\mathrm{m} = \frac{1}{N_p}\sum_{j=1}^{N_\mathrm{d}} |\partial u_i/\partial \mathrm{t} + u_i \cdot \nabla u_i + \nabla \mathrm{p}/\rho - \nu\nabla^2 u_i|^2$
$\mathcal{L} = \lambda_d\mathcal{L}_d + \lambda_i\mathcal{L}_i + \lambda_m\mathcal{L}_m$

Figure 1. *Schematic of the Physics-Informed Neural Network (PINN). The inputs consist of time t and spatial coordinates **x**, and the outputs are velocity **u** and pressure p. The neural network is trained using the continuity equation and the incompressible Navier–Stokes equations as physical constraints, in combination with the provided LPT dataset.*

2.1 Neural architecture

PINNs can adopt various neural network architectures depending on desired expressivity, computational efficiency, and implementation complexity. In this study, we use a Multi-Layer Perceptron (MLP), which is widely adopted and has shown robust performance in related literature.

Given a network of depth D, each hidden layer N_k is defined as:

$$N_k\left(z^{k-1}\right) = W^k z^{k-1} + b^k \tag{5}$$

where W^k and b^k are weights and biases of the k^{th} layer. The network's output is computed by a composition of layers and activation functions:

$$\hat{u}(z;\theta) = (N_k \circ \sigma \circ N_{k-1} \circ \cdots \sigma \circ N_1)(z) \tag{6}$$

where $\circ$ is the composition operator, $\theta = \{W^k, b^k\}_{k=1}^{D}$ represents all trainable parameters. For the current study, we use an MLP with 15 hidden layers, each with 300 neurons, and tanh as the activation function. We initialize weights using Glorot uniform initialization and biases with zeros.

Training deep networks can be challenging due to the high-dimensional parameter space, which may lead to instability or divergence. To promote stable convergence, we adopt several well-established techniques. One of the most widely used method is input normalization which rescale each coordinate of input between (-1,1),

$$x_{rescale} = 2\frac{x - x_{min}}{x_{max} - x_{min}} - 1 \tag{7}$$

where x_{min} and x_{max} are minimum and maximum values of x in the domain respectively. Another method implemented in the current network is weight normalization where a weight parameter of the neural network is expressed as

$$\mathrm{W} = \frac{g}{\|v\|}v \tag{8}$$

where v is a k-dimensional vector, g is a scalar, and $\|v\|$ denotes the Euclidean norm of v. Finally, to mitigate the neural network's tendency to prioritize learning low-frequency features from the training data (Cao et al.; S. Wang, Yu, et al. 2020)—an effect known as spectral bias, which results in overly smoothed fluid flow structures—we replace the first hidden layer with a Fourier Feature layer:

$$z^1 = (\sigma \circ N_1)(z^0) = [\sin(2\pi w_1 \cdot z^0), \cos(2\pi w_1 \cdot z^0), \cdots, \sin(2\pi w_w \cdot z^0), \cos(2\pi w_w \cdot z^0)]^T \tag{9}$$

where w_i are randomly chosen frequencies fixed before training (Tancik et al.; S. Wang, Wang, et al. 2020).

2.2 Optimization

The optimization of PINNs involves simultaneously fitting observed data and satisfying physical laws, making the problem inherently stiff and sensitive to hyperparameters. As network depth and dataset size increase, the risk of unstable convergence or divergence becomes more pronounced. To ensure stability in training, especially for large-scale LPT datasets, several techniques are incorporated.

Residual points, where the governing physical constraints are evaluated, play a role similar to mesh points in traditional numerical methods. Their placement critically influences how well the model satisfies physical laws. While uniform sampling is commonly used, we adopt a Residual-based Adaptive Distribution (RAD) method (Basir and Senocak, 2023), which dynamically resamples residual points according to a probability density function proportional to the PDE residual. The PDF is defined as:

$$\mathrm{p(x)} \propto \frac{\varepsilon^k(x)}{\mathbb{E}[\varepsilon^k(x)]} + \mathrm{c} \tag{10}$$

where ε(x) is the PDE residual, k and c are hyperparameters (set to 1 in this study), and $\mathbb{E}[\varepsilon^k(x)]$ is approximated using numerical integration with 10,000 integration steps. This method biases sampling toward regions where the physical constraints are most violated, improving physical consistency in the learned solution.

Another key challenge is balancing the multiple loss terms. If the data loss dominates, the network may overfit the observed particle data and fail to enforce physical constraints. Conversely, if the physics loss dominates, the solution may become overly smoothed or diverge from measurements. To improve convergence and overcome optimization challenges from competing loss directions, we employ the shampoo with adam in the preconditioner's eigenbasis (SOAP) optimizer (Vyas et al.; S. Wang, Bhartari, et al. 2024). This optimizer combines the adaptive nature of Adam with second-order curvature information from the local loss landscape. For a layer with weight matrix $W_t \in \mathbb{R}^{m\times n}$ and gradient $G_t \in \mathbb{R}^{m\times n}$, the optimizer computes:

$$L_t = \beta_2 L_{t-1} + (1-\beta_2)G_t G_t^T, \qquad R_t = \beta_2 R_{t-1} + (1-\beta_2)G_t^T G_t \tag{11}$$

where β_2 is exponential moving average coefficient. These matrices are decomposed as:

$$L_t = Q_L \Lambda_L Q_L^T, \qquad R_t = Q_R \Lambda_R Q_R^T \tag{12}$$

The gradient is projected into the eigenspace:

$$\tilde{G}_t = Q_L^T G_t Q_R \tag{13}$$

Adam updates are then performed in this rotated space:

$$\widetilde{W}_{t+1} = \widetilde{W}_t - \eta \cdot Adam(\tilde{G}_t) \tag{14}$$

$$W_{t+1} = Q_L \widetilde{W}_{t+1} Q_R^T \tag{15}$$

This method exploits curvature-aware updates to navigate the high-dimensional loss landscape more effectively than traditional first-order optimizers. Figure 2 demonstrates that the SOAP optimizer consistently improves convergence speed across all loss terms compared to the standard Adam optimizer.

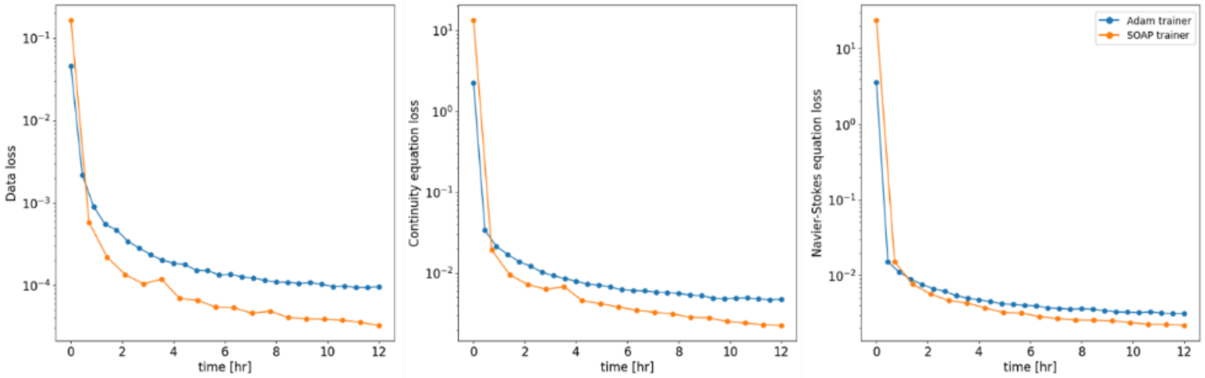


Figure 2*. Comparison of convergence behavior of data and physics loss between ordinary Adam optimizer and SOAP optimizer in training the PINN.*

3 Data description

We prepared three different cases: the first two use data from direct numerical simulations (DNS)—of a forced homogeneous isotropic turbulence (HIT) flow at Taylor microscale Re = 433 and a turbulent channel flow at $Re_\tau = 1000$. The third case uses experimental measurements of turbulent boundary layer flow as well at $Re_\tau = 995$ near the wall.

3.1 Synthetic particle tracks in DNS of homogeneous isotropic turbulent flow

The first synthetic dataset is based on DNS of homogeneous isotropic turbulence at Re = 433, available from the Johns Hopkins Turbulence Database (JHTDB). From the full simulation domain of size 2π×2π×2π (with 1024^3 voxels), a central sub-volume of 128^3 voxels were extracted. A total of 201 timesteps were stored, with a time interval of 0.002. Assuming air at 20°C as the working fluid, the integral length, time, and velocity scales are $l_0 = 127.32$ mm, $t_0 = 0.2\ s$, and $u_0 = 636.81\ mm/s$, respectively.

3.2 Synthetic particle tracks in DNS of turbulent channel flow

Similarly, synthetic particles were tracked in DNS of turbulent channel flow at a friction Reynolds number of Re_τ = 1000, also available from JHTDB. Using the viscous length scale $\delta_\nu = \nu/u_\tau$ (where ν is kinematic viscosity and u_τ is friction velocity), we extracted a subdomain of size $1546\delta_\nu \times 331\delta_\nu \times 117\delta_\nu$ in the streamwise, wall-normal, and spanwise directions, respectively, over 51 viscous time units. The working fluid was assumed to be air with a kinematic viscosity of 15 mm²/s and a free-stream velocity $u_\infty = 8.3m/s$. The corresponding viscous length and time scales were 0.036 mm and 0.09 ms, respectively. A pressure gradient of dp/dx = -0.0025, used in the DNS, was also considered during error evaluation.

3.3 Experimental particle tracks of turbulent boundary layer (TBL)

An experimental dataset was prepared under conditions similar to the synthetic DNS of turbulent channel flow. Measurements were conducted in the one-meter wind tunnel at DLR Göttingen (Schröder et al. 2024). The test section was 3000 mm long with a cross-section of 740 mm × 1000 mm. Air entered the test section at a free-stream velocity of $u_\infty = 10m/s$ and passed a trip device that initiated the development of a turbulent boundary layer along the lower wall. Measurements were taken through a window located 2.08 m downstream of the trip. The boundary layer had a friction Reynolds number of $Re_\tau = 995$, corresponding to a viscous length scale $l_\nu = 0.037mm$ and a viscous time scale $\tau_\nu =$ 0.0925ms.

Di-Ethyl-Hexyl-Sebacat (DEHS) particles with a mean diameter of approximately 1.5 µm were generated using a Laskin-nozzle seeder and homogeneously distributed in the tunnel. A dual-head nanosecond laser (Photonics Industries DM200) illuminated the particles from the downstream direction as Figure 3 (a). Images were captured through a bottom window using five high-speed cameras (four Phantom v2640, one v1840) with Scheimpflug adapters as depicted in Figure 1 (b), operating at 23.9 kHz and a spatial resolution of 34.185 px/mm. The measurement volume was 60 × 2 ×15 mm^3 in the streamwise, wall-normal, and spanwise directions, respectively.

Camera calibration was performed using a LaVision Type 11 3D calibration plate, followed by volume self-calibration and optical transfer function refinement. Particle image densities ranged from 0.06 to 0.07 ppp. The Shake-The-Box algorithm tracked 50,000–60,000 particles over up to 140,000 consecutive images.

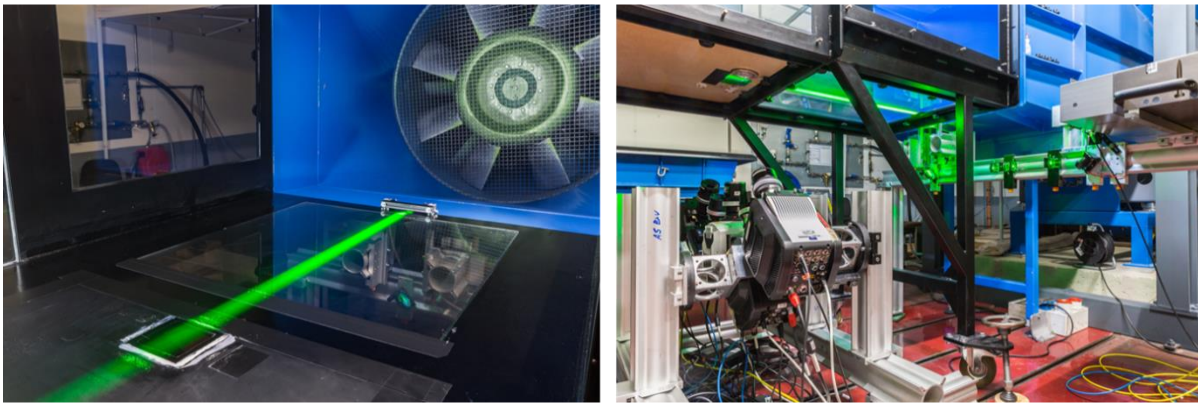
3.4 LPT dataset

For the synthetic datasets, particles were treated as Lagrangian tracers. Synthetic tracks were generated by simulating particle trajectories within a predefined inspection volume. Neutral buoyant particles were uniformly seeded throughout an extended domain and advected using a fourth-order Runge–Kutta scheme. To minimize boundary effects and ensure statistical convergence, the simulation domain was enlarged by approximately 15% beyond the interrogation volume in each spatial direction. However, only particles within the interrogation volume were retained for the final datasets. A total of 51 consecutive frames were prepared for data assimilation, containing between 50,000 and 280,000 particles per frame, depending on the case.

Similarly, for the experimental case, 51 consecutive time steps of measured particle trajectories were collected, with an average of 54,000 particles per frame. In the absence of ground truth, particles were divided into training and validation sets: 80% of the particle tracks were used to reconstruct the flow field, while the remaining 20% were withheld and used for error estimation.

The extracted particle trajectories were further processed using the B-spline-based curve fitting method TrackFit, which reduces noise by incorporating Wiener type optimal filtering (Gesemann et al. 2016). By regenerating the particle trajectories, first- and second-order derivatives can be computed at any time point within the observed time interval. The computed first derivatives of the trajectories, along with the corresponding spatiotemporal particle positions, are used as input for the PINN reconstruction.

To investigate the effect of particle sparsity, the training particles were downsampled by a factor of $k_n = 2^n$, for n=1,2, 3. This corresponds to mean inter-particle spacings ranging from $6l_\nu$ to $24l_\nu$ for the synthetic cases, and from $9.4l_\nu$ to $18.7l_\nu$ for the experimental case.



(a) (b)

Figure 3. *Experimental setup in the 1-meter wind tunnel at DLR Göttingen. (a) Illumination of the measurement volume near the wall using two Photonics Industries DM200 lasers, with indications of the flow direction and imaged region. (b) Five high-speed cameras—four Phantom v2640 and one Phantom v1840—record particle light scattering within the measurement volume through an optical window.*

4 Results

4.1 PINN reconstruction result at 24th timestep

Figure 4 shows the velocity and pressure reconstruction results from the PINN applied to the HIT case, comparing a densely seeded scenario (l_ν=6) with a more sparsely seeded one (l_ν=15.1). The corresponding error plots, compared against DNS ground truth, are also presented. In the densely seeded case, all velocity components are accurately reconstructed, and the velocity gradients are well preserved, as evidenced by the high-quality pressure field. In the sparsely seeded case, although the PINN successfully captures the overall flow structure, the reduced number of particles in key regions results in a smoother velocity field. This smoothing causes some loss of velocity gradient information, leading to a slightly less accurate pressure reconstruction. Figure 5 illustrates the coherent structures of the HIT flow using a Q-criterion threshold of 3,500s^{-1}. For inter-particle spacings up to 9.5$l_\nu$most fine-scale coherent structures are well preserved. However, as the spacing increases, the PINN begins to miss finer structures, and under the sparsest conditions, even some of the larger flow structures are no longer accurately reconstructed.

Figure 6 shows the reconstruction results from the PINN applied to the synthetic turbulent boundary layer case at two different particle seeding densities. In contrast to the HIT case, the PINN effectively reconstructs the velocity field and preserves velocity gradients even under relatively sparse seeding, despite having similar inter-particle spacing in terms of viscous length scale. This result is further supported by Figure 7, which visualizes the coherent structures of the synthetic TBL flow at a Q-criterion of 2.5×10^6 s^{-1}. Unlike in the HIT case, the PINN maintains the velocity gradients with greater consistency across different particle densities. Although some fine-scale features are missing at the lowest density, the main coherent structures remain largely intact, indicating robustness to sparsity in the training data.

Figure 8 presents slice plots of each velocity component and the pressure field from the experimental TBL case. Due to the absence of ground truth DNS data, only directional slice plots are shown. Nevertheless, the reconstructed velocity and pressure fields appear consistent across different inter-particle spacings. Notably, the pressure field shows no evidence of filtering or degradation, even at lower particle concentrations. The robustness of the experimental TBL reconstruction is further demonstrated in Figure 9, which shows iso-surfaces of coherent structures based on a Q-criterion of 1.25×10^6. Even at the lowest particle density, fine-scale structures remain well preserved.

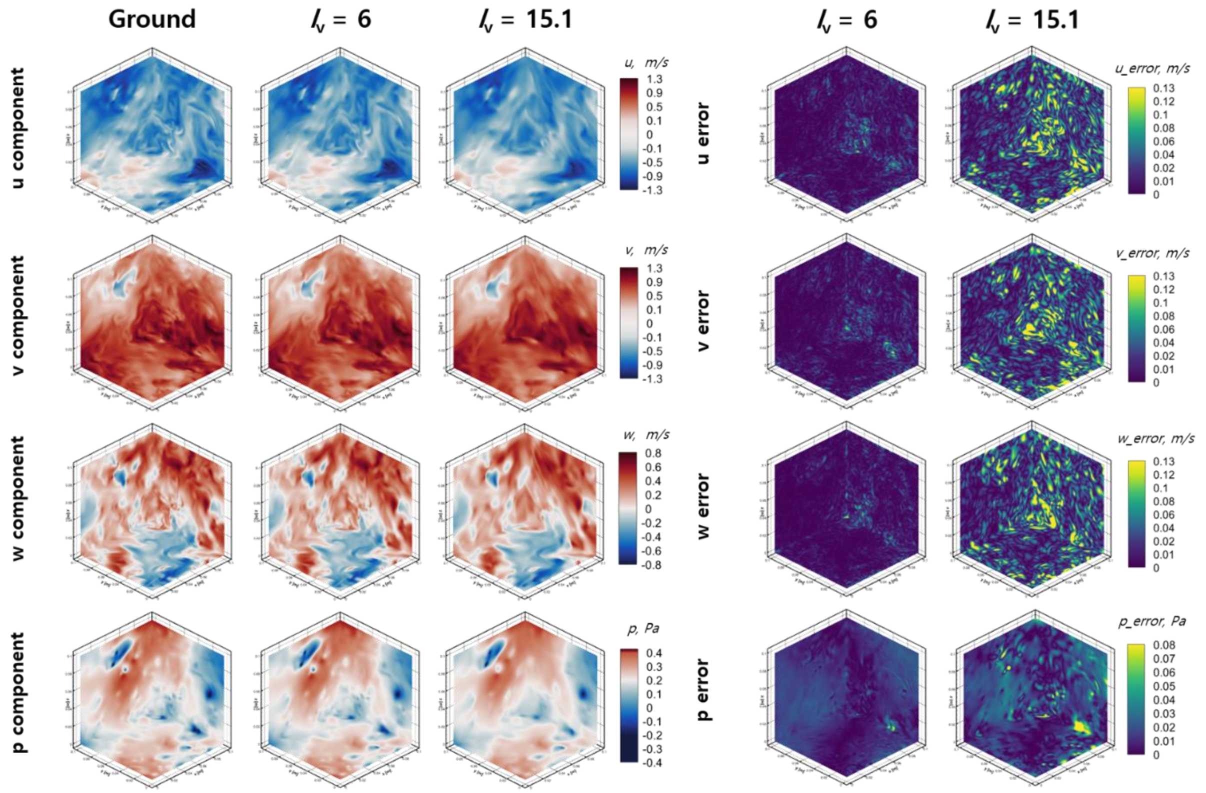


Figure 4. *PINN reconstruction of velocity and pressure fields for the synthetic HIT case at different inter-particle spacings (6$\boldsymbol{l_\nu}$ and 15.1$\boldsymbol{l_\nu}$), along with corresponding error plots at the 24th timestep.*

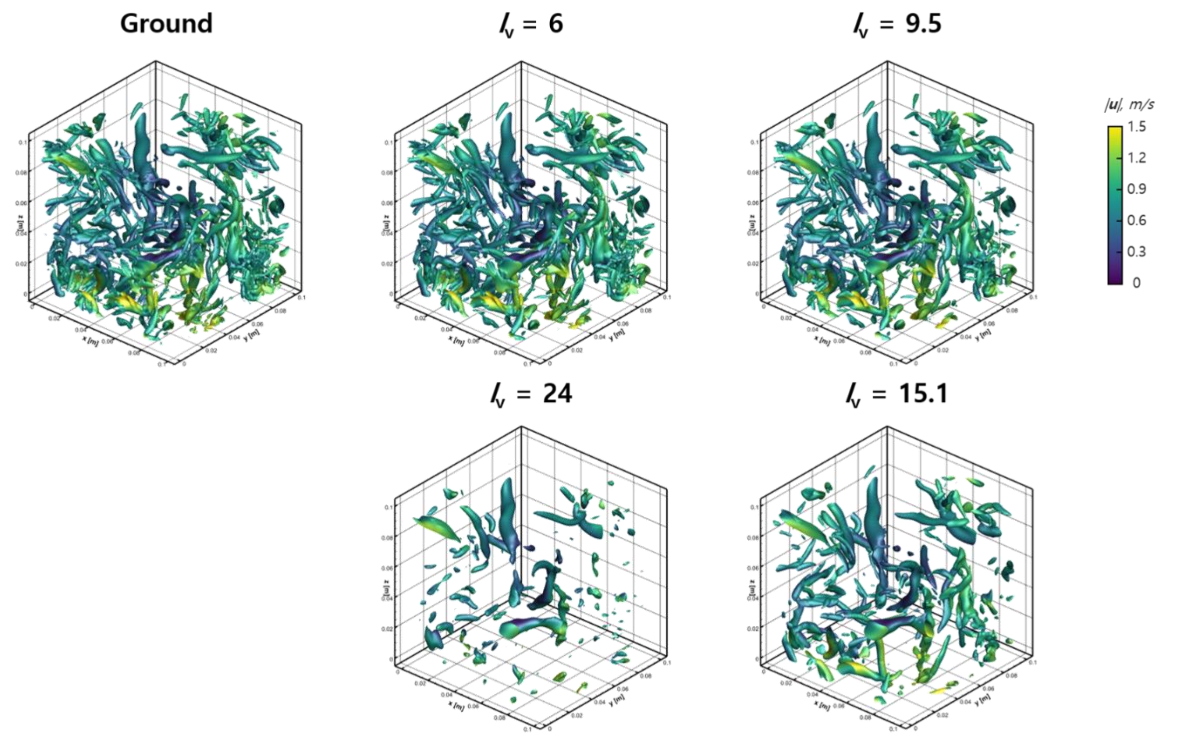


Figure 5. *Visualized coherent structures using isosurface of the Q-criterion field (Q = 3500 s^{-2}) in synthetic HIT flow reconstructed by PINN across different inter-particle spacing at the 24th timestep.*

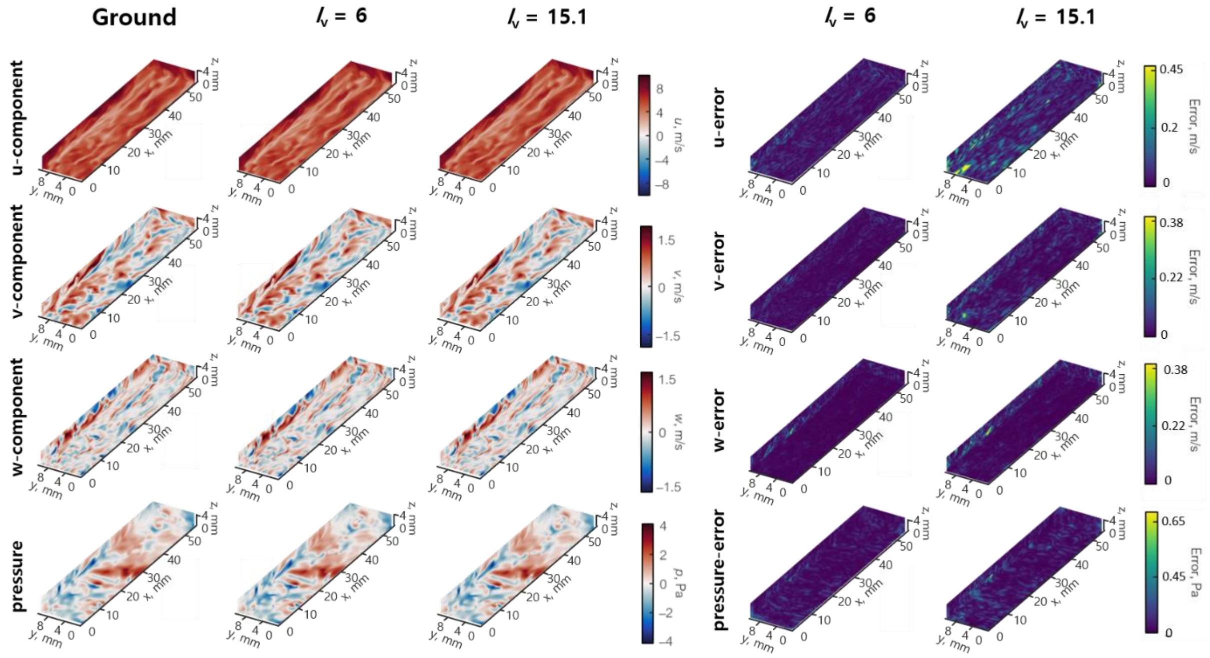


Figure 6. *PINN reconstruction of velocity and pressure fields for the synthetic TBL case at different inter-particle spacings (6$\mathbf{l_v}$ and 15.1$\mathbf{l_v}$), along with corresponding error plots at the 24th timestep.*

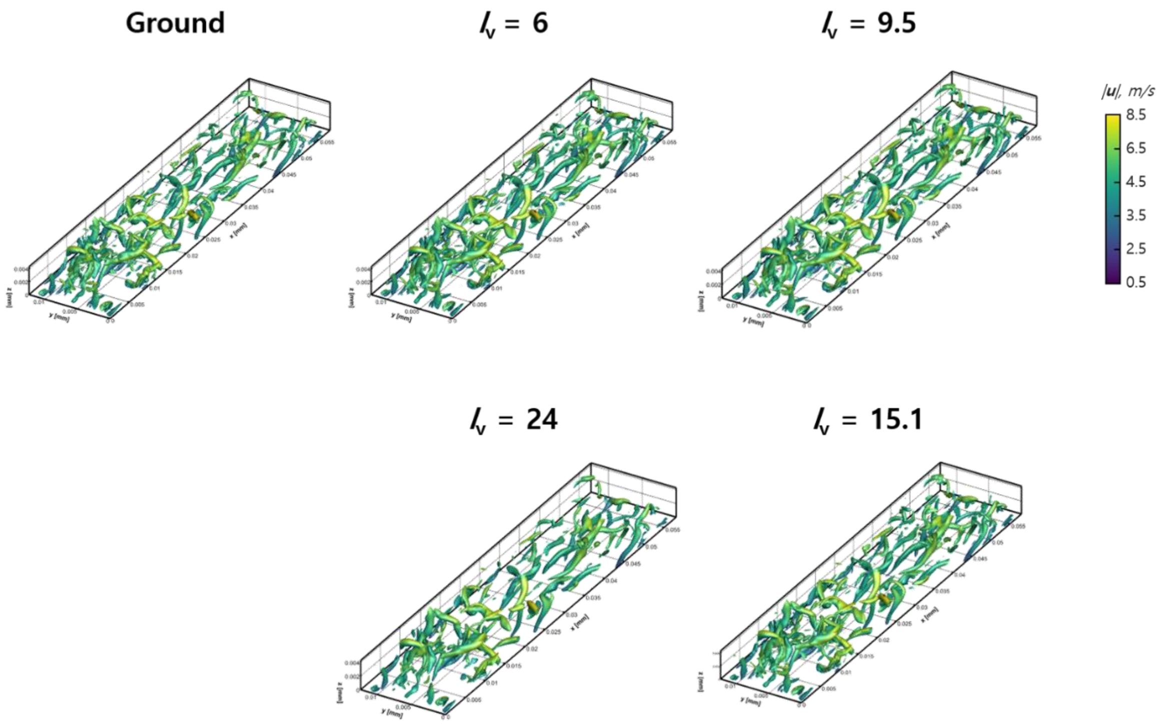


Figure 7. *Visualized coherent structures using isosurface of the Q-criterion field ($Q = 2.5 \times 10^6\ s^{-2}$) in synthetic TBL flow reconstructed by PINN across different inter-particle spacing at the 24th timestep.*

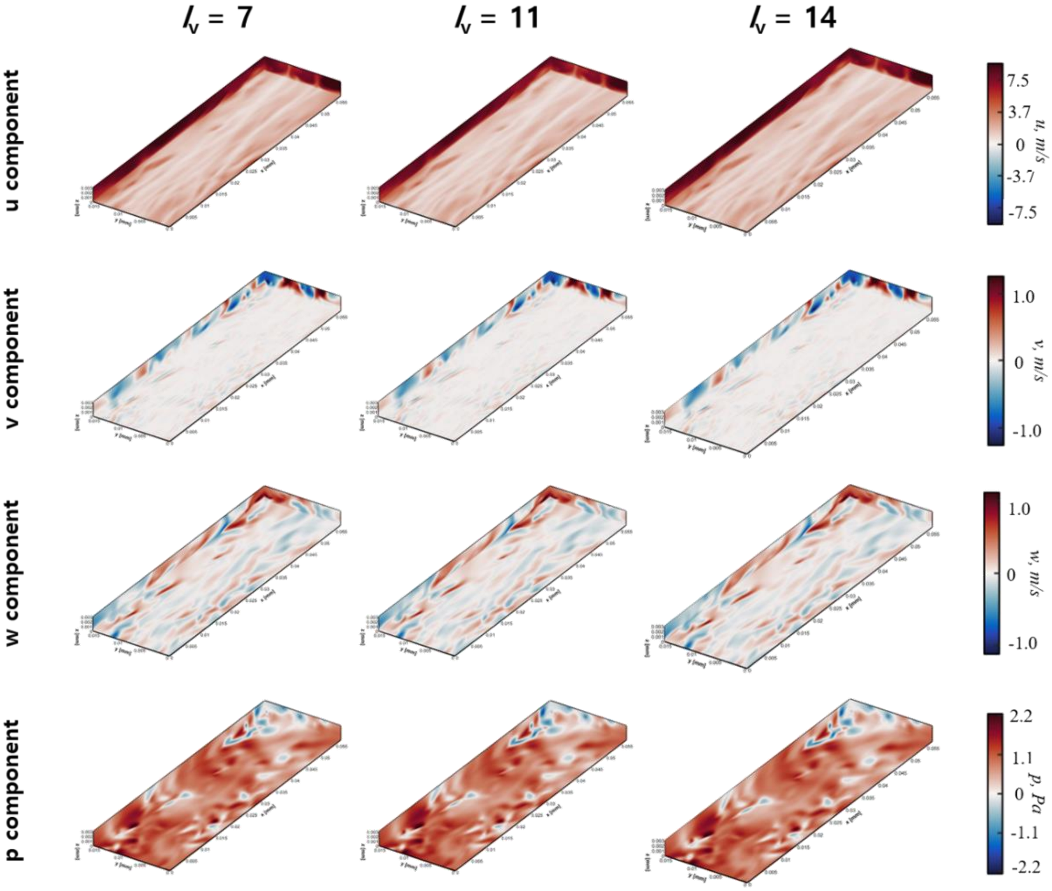


Figure 8. *PINN reconstruction of velocity and pressure fields for the experimental TBL case at different inter-particle spacings ($7\mathbf{l_v}$, $11\mathbf{l_v}$, and $14\mathbf{l_v}$), along with corresponding error plots at the 24^{th} timestep.*

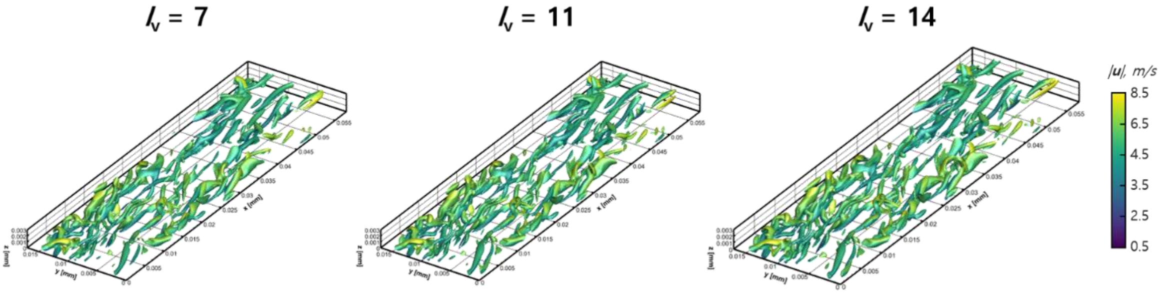


Figure 9. *Visualized coherent structures using isosurface of the Q-criterion field ($Q = 1.25\times10^6\ s^{-2}$) in experimental TBL flow reconstructed by PINN across different inter-particle spacing at the 24th timestep.*

4.2 Error evaluation

Figure 10 shows the normalized root mean square error (NRMSE) across all test cases as a function of seeding density. For the synthetic HIT and TBL cases, where ground truth velocity and pressure fields are available, the PINN outputs are directly compared to the DNS data. In the experimental TBL case, where no ground truth is available, we evaluate performance using the 20% of particle trajectories that were excluded from training. Velocity error is estimated from these withheld trajectories, while acceleration is evaluated from the second-order derivatives obtained via TrackFit reconstruction. The NRMSE is computed using the following metric:

$$e_\varphi = \left(\frac{<\|\varphi_{pred} - \varphi_{exact}\|_2^2>}{<\|\varphi_{exact}\|_2^2>}\right)^{\frac{1}{2}}, \qquad <\varphi>= \frac{1}{|\Omega|}\int \varphi(x)ds \tag{16}$$

For velocity error, we compute the fluctuation velocity error by subtracting the mean velocity from each sample. The results show that NRMSE generally increases with inter-particle distance. However, in contrast to the HIT case—which is highly sensitive to particle sparsity—both the synthetic and experimental TBL cases exhibit significantly greater robustness, maintaining relatively low error even at larger inter-particle spacings.

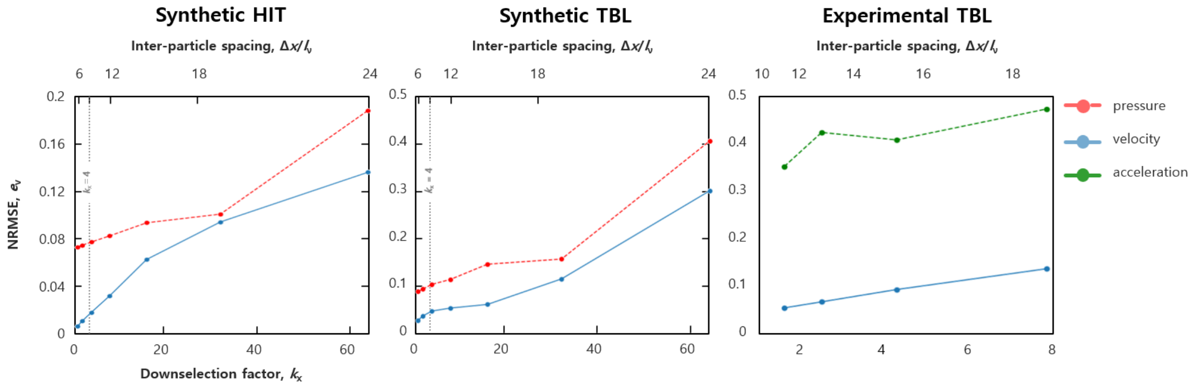


Figure 10. *NRMSE of PINN results for all test cases*

5 Conclusions

This study highlights PINNs' effectiveness for reconstructing detailed flow fields from sparse and dense LPT datasets. PINNs consistently preserve velocity gradients and pressure fields by enforcing incompressible Navier-Stokes equations. While HIT reconstructions were sensitive to particle spacing, TBL reconstructions—both synthetic and experimental—showed remarkable robustness, maintaining key flow features even at lower densities. Advanced optimization methods, such as SOAP, adaptive residual sampling, and neural tangent kernel-based loss balancing, were crucial for enhancing stability and accuracy. Overall, PINNs combined with these optimizations provide a robust and accurate framework for flow reconstruction from sparse measurements.

References

Agüera, Nereida, et al. "Ensemble 3D PTV for High Resolution Turbulent Statistics." *Measurement Science and Technology*, vol. 27, no. 12, Dec. 2016, p. 124011. *DOI.org (Crossref)*, https://doi.org/10.1088/0957-0233/27/12/124011.

Basir, Shamsulhaq, and Inanc Senocak. *An Adaptive Augmented Lagrangian Method for Training Physics and Equality Constrained Artificial Neural Networks*. arXiv:2306.04904, arXiv, 15 July 2023. *arXiv.org*, https://doi.org/10.48550/arXiv.2306.04904.

Cai, Shengze, et al. "Physics-Informed Neural Networks Enhanced Particle Tracking Velocimetry: An Example for Turbulent Jet Flow." *IEEE Transactions on Instrumentation and Measurement*, vol. 73, 2024, pp. 1–9. *DOI.org (Crossref)*, https://doi.org/10.1109/TIM.2024.3398068.

Cao, Yuan, et al. *Towards Understanding the Spectral Bias of Deep Learning*. arXiv:1912.01198, arXiv, 5 Oct. 2020. *arXiv.org*, https://doi.org/10.48550/arXiv.1912.01198.

Clark Di Leoni, Patricio, et al. "Reconstructing Turbulent Velocity and Pressure Fields from Under-Resolved Noisy Particle Tracks Using Physics-Informed Neural Networks." *Experiments in Fluids*, vol. 64, no. 5, May 2023, p. 95. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-023-03629-4.

Gesemann, Sebastian, et al. From Noisy Particle Tracks to Velocity, Acceleration and Pressure Fields Using B-Splines and Penalties. *Proceedings of the International Symposium on the Application of Laser and Imaging Tech-niques to Fluid Mechanics*, July 2016, Lisbon, Portugal,

Godbersen, Philipp, et al. "FlowFit3: Efficient Data Assimilation Of LPT Measurements." *Proceedings of the International Symposium on the Application of Laser and Imaging Techniques to Fluid Mechanics*, vol. 21, July 2024, pp. 1–14. *DOI.org (Crossref)*, https://doi.org/10.55037/lxlaser.21st.216.

Jeon, Young Jin, et al. "Fine Scale Reconstruction (VIC#) by Implementing Additional Constraints and Coarse-Grid Approximation into VIC+." *Experiments in Fluids*, vol. 63, no. 4, Apr. 2022, p. 70. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-022-03422-9.

Kasagi, Nobuhide, and Koichi Nishino. "Probing Turbulence with Three-Dimensional Particle-Tracking Velocimetry." *Experimental Thermal and Fluid Science*, vol. 4, no. 5, Sept. 1991, pp. 601–12. *DOI.org (Crossref)*, https://doi.org/10.1016/0894-1777(91)90039-T.

Raissi, M., et al. "Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations." *Journal of Computational Physics*, vol. 378, Feb. 2019, pp. 686–707. *DOI.org (Crossref)*, https://doi.org/10.1016/j.jcp.2018.10.045.

Schanz, Daniel, et al. "Shake-The-Box: Lagrangian Particle Tracking at High Particle Image Densities." *Experiments in Fluids*, vol. 57, no. 5, May 2016, p. 70. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-016-2157-1.

Schneiders, Jan F. G., et al. "Time-Supersampling of 3D-PIV Measurements with Vortex-in-Cell Simulation." *Experiments in Fluids*, vol. 55, no. 3, Mar. 2014, p. 1692. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-014-1692-x.

Schneiders, Jan F. G., and Fulvio Scarano. “Dense Velocity Reconstruction from Tomographic PTV with Material Derivatives.” *Experiments in Fluids*, vol. 57, no. 9, Sept. 2016, p. 139. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-016-2225-6.

Schröder, Andreas, et al. “Near-Wall Flow Features In ZPG-TBL At Various Reynolds Numbers Using Dense 3D Lagrangian Particle Tracking.” *Proceedings of the International Symposium on the Application of Laser and Imaging Techniques to Fluid Mechanics*, vol. 21, July 2024, pp. 1–16. *DOI.org (Crossref)*, https://doi.org/10.55037/lxlaser.21st.226.

Schröder, Andreas, and Daniel Schanz. “3D Lagrangian Particle Tracking in Fluid Mechanics.” *Annual Review of Fluid Mechanics*, vol. 55, no. 1, Jan. 2023, pp. 511–40. *DOI.org (Crossref)*, https://doi.org/10.1146/annurev-fluid-031822-041721.

Sciacchitano, A., et al. “On the Accuracy of Data Assimilation Algorithms for Dense Flow Field Reconstructions.” *Experiments in Fluids*, vol. 66, no. 2, Feb. 2025, p. 42. *DOI.org (Crossref)*, https://doi.org/10.1007/s00348-025-03969-3.

Steinfurth, B., et al. “Physics-Informed Neural Networks for Dense Reconstruction of Vortex Rings from Particle Tracking Velocimetry.” *Physics of Fluids*, vol. 36, no. 9, Sept. 2024, p. 095110. *DOI.org (Crossref)*, https://doi.org/10.1063/5.0212585.

Tancik, Matthew, et al., *Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains*. in Advances in Neural Information Processing Systems 33 (NeurIPS 2020), https://proceedings.neurips.cc/paper_files/paper/2020/file/55053683268957697aa39fba6f231c68-Paper.pdf

Toscano, Juan Diego, et al. *Inferring Turbulent Velocity and Temperature Fields and Their Statistics from Lagrangian Velocity Measurements Using Physics-Informed Kolmogorov-Arnold Networks*. arXiv:2407.15727, arXiv, 23 July 2024. *arXiv.org*, https://doi.org/10.48550/arXiv.2407.15727.

Vyas, Nikhil, et al. *SOAP: Improving and Stabilizing Shampoo Using Adam*. arXiv:2409.11321, arXiv, 31 Jan. 2025. *arXiv.org*, https://doi.org/10.48550/arXiv.2409.11321.

Wang, Hongping, et al. “Dense Velocity Reconstruction from Particle Image Velocimetry/Particle Tracking Velocimetry Using a Physics-Informed Neural Network.” *Physics of Fluids*, vol. 34, no. 1, Jan. 2022, p. 017116. *DOI.org (Crossref)*, https://doi.org/10.1063/5.0078143.

Wang, Sifan, Ananyae Kumar Bhartari, et al. *Gradient Alignment in Physics-Informed Neural Networks: A Second-Order Optimization Perspective*. arXiv:2502.00604, arXiv, 2 Feb. 2025. *arXiv.org*, https://doi.org/10.48550/arXiv.2502.00604.

Wang, Sifan, Hanwen Wang, et al. “On the Eigenvector Bias of Fourier Feature Networks: From Regression to Solving Multi-Scale PDEs with Physics-Informed Neural Networks.” *Computer Methods in Applied Mechanics and Engineering*, vol. 384, Oct. 2021, p. 113938. *DOI.org (Crossref)*, https://doi.org/10.1016/j.cma.2021.113938.

Wang, Sifan, Xinling Yu, et al. “When and Why PINNs Fail to Train: A Neural Tangent Kernel Perspective.” *Journal of Computational Physics*, vol. 449, Jan. 2022, p. 110768. *DOI.org (Crossref)*, https://doi.org/10.1016/j.jcp.2021.110768.