Physics-Informed Interpolation of Velocity and Pressure Gradient from Particle Tracking Velocimetry Data

Yang Zhang [1,*], Mahyar Moaven [1], Douglas Carter [1], Louis Cattafesta [1], Peter Schmid[2]

1: Department of Mechanical, Materials, and Aerospace Engineering, Illinois Institute of Technology, US
2: Physical Science and Engineering Division, King Abdullah University of Science and Technology, Saudi Arabia
*Corresponding author: yzhang215@illinoistech.edu

Keywords: PTV, Physics-Informed, Data assimilation, Eulerian Mapping

ABSTRACT

We introduce a physics-informed local interpolation framework for reconstructing velocity, velocity gradients, velocity Hessians, and pressure gradients on structured grids from particle-based measurements, including particle positions, velocities, and material accelerations. The method embeds the incompressible Navier–Stokes momentum equation as linear constraints within a weighted least-squares formulation, ensuring physically consistent estimates from irregularly distributed PTV data. Validation using direct numerical simulation (DNS) data from the JHU turbulence database demonstrates accurate reconstruction of velocity fields, vorticity, and pressure gradients, including coherent structures and near-wall features. Experimental results from time-resolved PTV of a turbulent water jet further highlight the method's robustness, enabling high-fidelity 3D flow-field reconstruction and subsequent modal analysis via POD. The approach provides a unified and robust tool for mapping Lagrangian particle measurements to Eulerian representations suitable for gradient-based diagnostics and pressure reconstruction.

1. Introduction

Particle Image Velocimetry (PIV) produces structured, grid-based velocity fields, enabling direct computation of flow statistics and differential quantities. In contrast, Particle Tracking Velocimetry (PTV) data consist of scattered spatial samples, which complicates conventional numerical operations, especially spatial differentiation. This limitation has led to extensive development of mapping strategies, including data-assimilation approaches such as VIC+ (Gesemann et al., 2016) and VIC# (Jeon et al., 2022), which reconstruct dense velocity fields from sparse particle trajectories. Other efforts employ mesh-free interpolation techniques such as radial basis functions (Fornberg & Flyer, 2015) and affine gradient fitting (Fenelon et al., 2025). While the globally coupled methods offer physically regularized reconstructions, they rely on repeated elliptic solves or nonlinear optimization over the full domain, which can become computationally expensive as spatial resolution

increases. In parallel, physics-informed neural networks (PINNs) (Raissi et al., 2019) have also embedded the Navier-Stokes equation into the neural network training for flow reconstruction. While flexible and capable of incorporating sparse data, PINN-based approaches require nonlinear optimization over high-dimensional parameter spaces and typically involve substantial training time, especially for three-dimensional time-resolved datasets.

Building on our earlier Taylor-series-based (TS) work for computing velocity gradients and Hessians directly at particle locations (Fenelon et al., 2025; Zhang et al., 2025), we extend this TS formulation to perform physics-informed interpolation onto arbitrary target locations, such as structured Eulerian grids. The TS representation provides a robust alternative to element-based affine fitting methods, which can suffer from sensitivity to particle-group geometry (Fenelon et al., 2025). By embedding the local form of the momentum equation as linear constraints, the method simultaneously reconstructs velocity derivatives and pressure gradients using only PTV-measured positions, velocities, and material accelerations. Key enhancements over the preliminary conference version (Zhang et al., 2025) include an exact divergence-free constraint via KKT multipliers, Huber-loss robust estimation with IRLS, and systematic noise sensitivity analysis. The method scales linearly with the number of evaluation points and requires no global Poisson solves, neural-network training, or iterative domain-wide optimization.

The remainder of the paper is organized as follows. Section 2 presents the interpolation methodology and its formulation. Section 3 validates the method using DNS channel-flow from the Johns Hopkins University turbulence database (Graham et al., 2014) and demonstrates applications to experimental time-resolved PTV measurements from LaVision (LaVision GmbH, 2026). The conclusions are given in the end.

2. Methodology

2.1. Taylor-Series Formulation and Momentum Constraints

Given a set of Lagrangian particle measurements $\{(\mathbf{x}_k,\ \mathbf{u}_k,\ \mathbf{a}_k)\}_{k=1}^{N_p}$, the objective is to compute the velocity field, velocity gradients (& Hessians), and pressure gradients at a set of target (structured grid) points $\{\mathbf{x}_i^\star\}_{i=1}^{N_s}$. At each target $\mathbf{x}^\star$, the velocity and pressure are modeled using local second-order Taylor series expansions. For each target and particle pair, we define offsets $\Delta\mathbf{x}_k = \mathbf{x}_k - \mathbf{x}^\star$.

For each velocity component u_c ($c = 1, 2, 3$) define

$$\mathbf{u}_c(\mathbf{x}^\star + \Delta\mathbf{x}) \approx \mathbf{u}_c(\mathbf{x}^\star) + \mathbf{b}_c^\top \Delta\mathbf{x} + \tfrac{1}{2}\,\Delta\mathbf{x}^\top H^{(u_c)} \Delta\mathbf{x}, \tag{1}$$

where $\mathbf{b}_c \in \mathbb{R}^3$ are spatial derivatives (row c of $\nabla\mathbf{u}$), and symmetric $H^{(u_c)}$ has 6 independent second-order coefficients. Each velocity component carries 10 unknowns (one velocity value, three first-order derivatives, and six second-order derivatives), giving $n_{\text{vel}} = 30$ for all three compo-

nents. The pressure gradient contributes 3 additional unknowns $\nabla p(\mathbf{x}) \approx \mathbf{g}_p$, for a total of $n = 33$ unknowns.

The incompressible Navier–Stokes momentum equation provides the link between measured material accelerations and pressure:

$$\rho \frac{D\mathbf{u}}{Dt}(\mathbf{x}) = -\nabla p(\mathbf{x}) + \mu \nabla^2 \mathbf{u}(\mathbf{x}). \tag{2}$$

For a neighbor particle at $\mathbf{x}_k = \mathbf{x}^\star + \Delta\mathbf{x}_k$ with measured acceleration $\mathbf{a}_k \equiv D\mathbf{u}/Dt|_k$, the Laplacian simplifies to $\nabla^2 u_c(\mathbf{x}_k) \approx \mathrm{tr}\left(H^{(u_c)}\right)$ under the second-order truncation. The momentum constraint for neighbor k becomes three scalar equations:

$$\rho\, \mathbf{a}_k = -\mathbf{g}_p + \mu \begin{bmatrix} \mathrm{tr}(H^{(u_1)}) \\ \mathrm{tr}(H^{(u_2)}) \\ \mathrm{tr}(H^{(u_3)}) \end{bmatrix}. \tag{3}$$

2.2. Weighted Least-Square System

Collect neighbor equations for a target $\mathbf{x}^\star$ using m nearest neighbors. Each neighbor contributes:

- *Velocity reproduction* equations: 3 scalar equations (Eq. (1)),
- *Momentum* equations: 3 scalar equations (Eq. (3)).

Therefore, the total number of equations is $6m$.

Form the weighted linear system

$$\underbrace{A}_{(6m)\times n}\ \theta\ =\ \underbrace{b}_{(6m)\times 1},$$

where θ is the stacked vector of all unknown coefficients (velocity derivatives/Hessians, pressure gradients), and n is the number of unknowns.

Below we give an explicit block construction for the common case: *velocity derivatives* (27 unknowns) and *pressure gradients* (3 unknowns), so $n = 30$.

For each neighbor k define the 10-term velocity derivative basis:

$$\Phi^{(10)}(\Delta\mathbf{x}_k) = \begin{bmatrix} 1 \\ \Delta x_k \\ \Delta y_k \\ \Delta z_k \\ \frac{1}{2}\Delta x_k^2 \\ \Delta x_k \Delta y_k \\ \Delta x_k \Delta z_k \\ \frac{1}{2}\Delta y_k^2 \\ \Delta y_k \Delta z_k \\ \frac{1}{2}\Delta z_k^2 \end{bmatrix} \in \mathbb{R}^{10}.$$

For neighbor k, component 1 (x-velocity) produces equation:

$$\Phi^{(10)}(\Delta\mathbf{x}_k)^\top \theta_{u1} = u_{k,1}.$$

Similarly for components 2 and 3. Collecting all m neighbors yields the $3m$ rows:

$$A_{\text{vel}} = \begin{bmatrix} \Phi^{(10)}(\Delta\mathbf{x}_1)^\top & 0 & 0 \\ 0 & \Phi^{(10)}(\Delta\mathbf{x}_1)^\top & 0 \\ 0 & 0 & \Phi^{(10)}(\Delta\mathbf{x}_1)^\top \\ \vdots & \vdots & \vdots \\ \Phi^{(10)}(\Delta\mathbf{x}_m)^\top & 0 & 0 \\ 0 & \Phi^{(10)}(\Delta\mathbf{x}_m)^\top & 0 \\ 0 & 0 & \Phi^{(10)}(\Delta\mathbf{x}_m)^\top \end{bmatrix}_{3m \times n_{\text{vel}}}, \qquad b_{\text{vel}} = \begin{bmatrix} u_{1,1} \\ u_{1,2} \\ u_{1,3} \\ \vdots \\ u_{m,1} \\ u_{m,2} \\ u_{m,3} \end{bmatrix}_{3m \times 1}, \tag{4}$$

Stacked per component the velocity model provides 3 blocks of 10 unknowns: $\theta_{u1}, \theta_{u2}, \theta_{u3}$ (each 10×1). Pressure gradient unknowns $\theta_p = [g_x, g_y, g_z]^\top$ occupy 3 columns after the velocity block. The full unknowns vector becomes

$$\theta = \left[\theta_{u_1}^\top,\ \theta_{u_2}^\top,\ \theta_{u_3}^\top,\ \theta_p^\top\right]^\top \in \mathbb{R}^{33}.$$

The Laplacian terms can be extracted from the velocity Hessian coefficients as $\text{tr}(H^{(u_c)}) = \theta_{u_c,5} + \theta_{u_c,8} + \theta_{u_c,10}$.

Thus, the row for x-momentum (neighbor k) is:

$$-\mu \cdot [\,0\,0\,0\,0\;\;1\,0\,0\;\;1\,0\,1\,] \cdot \theta_{u1} + [\,1\,0\,0\,] \cdot \mathbf{g}_p = \; -\rho a_{k,1}. \tag{5}$$

Analogous patterns hold for y and z momentum rows. Stack all momentum rows to form A_{mom} and b_{mom}.

Because the distribution of neighbor particles is random, the contribution of information from each particle should be weighted based on the distance to the target location (e.g., inverse-distance-squared (ID2)). Therefore, a diagonal weight matrix W is applied to emphasize nearby neighbors:

$$W = \mathrm{blkdiag}(\sqrt{w_1} I_3, \ldots, \sqrt{w_m} I_3),$$

where I_3 is the 3×3 identity matrix. These weights are applied to both velocity and momentum blocks as

$$A = \begin{bmatrix} W_3 A_{\mathrm{vel}} \\ W_3 A_{\mathrm{mom}} \end{bmatrix}_{6m \times n}, \qquad b = \begin{bmatrix} W_3 b_{\mathrm{vel}} \\ W_3 b_{\mathrm{mom}} \end{bmatrix}_{6m \times 1}. \tag{6}$$

Then solve the regularized least-squares

$$\hat{\theta} = \arg\min_{\theta} \|A\theta - b\|_2^2 + \lambda \|I\theta\|_2^2,$$

where I is the identity matrix. Practically we append Tikhonov regularization rows:

$$A \leftarrow \begin{bmatrix} A \\ \sqrt{\lambda_{\mathrm{vel}}}\, I_{\mathrm{vel}} \\ \sqrt{\lambda_p}\, I_p \end{bmatrix}, \qquad b \leftarrow \begin{bmatrix} b \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

with λ_{vel} and λ_p being regularization coefficients. This regularizes subblocks independently.

2.3. Divergence-Free Constraint

For incompressible flows, the velocity field satisfies the continuity equation $\nabla \cdot \mathbf{u} = 0$. This constraint is enforced exactly using a Lagrange multiplier within a Karush–Kuhn–Tucker (KKT) system. From the Taylor expansion (1), the divergence at the target $\mathbf{x}^\star$ is determined by the diagonal elements of the velocity gradient tensor:

$$b_{1,1} + b_{2,2} + b_{3,3} = 0,$$

where $b_{c,j}$ denotes $\partial u_c / \partial x_j$. Defining the constraint row $C \in \mathbb{R}^{1 \times n}$ that selects these entries:

$$C = [\underbrace{0,\, 1,\, 0,\, \ldots,\, 0}_{10},\ \underbrace{0,\, 0,\, 1,\, 0,\, \ldots,\, 0}_{10},\ \underbrace{0,\, 0,\, 0,\, 1,\, 0,\, \ldots,\, 0}_{10},\ \underbrace{0,\, \ldots,\, 0}_{n_p}],$$

where $n_p = 3$ is the number of pressure-gradient unknowns. By imposing $C\theta = 0$, the continuity constraint is enforced exactly at the target point.

2.4. Regularization

Block-diagonal Tikhonov regularization is applied independently to the velocity-derivative and pressure-gradient unknowns, with the velocity values $u_c(\mathbf{x}^\star)$ left unregularized:

$$\Lambda = \mathrm{diag}\Big(0, \underbrace{\lambda_{\mathrm{vel}}, \ldots, \lambda_{\mathrm{vel}}}_{9}, 0, \underbrace{\lambda_{\mathrm{vel}}, \ldots, \lambda_{\mathrm{vel}}}_{9}, 0, \underbrace{\lambda_{\mathrm{vel}}, \ldots, \lambda_{\mathrm{vel}}}_{9}, \lambda_p, \lambda_p, \lambda_p\Big) \in \mathbb{R}^{n\times n}, \tag{7}$$

with $\lambda_{\mathrm{vel}} = \lambda_p = 10^{-10}$ throughout. This provides weak numerical stabilization without materially affecting the data-fitting or momentum-constraint terms. The full constrained problem,

$$\min_{\theta} \ \|A\theta - b\|^2 + \theta^\top \Lambda \theta \quad \text{subject to} \quad C\theta = 0, \tag{8}$$

is solved via the KKT system

$$\begin{bmatrix} A^\top A + \Lambda & C^\top \\ C & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \mu \end{bmatrix} = \begin{bmatrix} A^\top b \\ 0 \end{bmatrix}, \tag{9}$$

where $\mu \in \mathbb{R}$ is the Lagrange multiplier. The solution satisfies $C\theta = 0$ to machine precision ($|\nabla \cdot \mathbf{u}| < 10^{-12}$). When the divergence-free constraint is not required, the problem reduces to the standard form $\theta = (A^\top A + \Lambda)^{-1} A^\top b$.

2.5. Robust Estimation with Huber Loss

Experimental PTV data can contain spurious particle tracks due to measurement noise, occlusion, or tracking errors. To achieve robustness to outliers, the Huber loss function is employed (Ghosh et al., 2024) via Iteratively Reweighted Least Squares (IRLS). At iteration j, the following weighted problem is solved:

$$\theta^{(j+1)} = \arg\min_{\theta} \sum_{i=1}^{6m} w_i^{(j)} r_i^2(\theta) + \theta^\top \Lambda \theta \quad \text{subject to} \quad C\theta = 0 \tag{10}$$

where the weights are updated based on the residuals from the previous iteration:

$$w_i^{(j+1)} = \begin{cases} 1 & \text{if } |r_i^{(j)}| \le \delta, \\ \delta \,/\, |r_i^{(j)}| & \text{if } |r_i^{(j)}| > \delta. \end{cases} \tag{11}$$

where $r_i^{(j)}$ are the residuals from the previous iteration and the threshold δ is set automatically at each target point using the Median Absolute Deviation of the initial L_2 residuals: $\delta = 1.5 \times 1.4826 \times \mathrm{MAD}(r^{(0)})$. When no outliers are present, the Huber solution converges to the L_2-solution within

numerical tolerance. When the divergence-free constraint is active, the KKT system is solved at each IRLS iteration with the updated weight matrix as

$$\begin{bmatrix} A^\top W^{(j)} A + \Lambda & C^\top \\ C & 0 \end{bmatrix} \begin{bmatrix} \theta^{(j+1)} \\ \mu^{(j+1)} \end{bmatrix} = \begin{bmatrix} A^\top W^{(j)} b \\ 0 \end{bmatrix}. \tag{12}$$

where $W^{(j)} = \mathrm{diag}(w_1^{(j)}, \ldots, w_{6m}^{(j)}) \in \mathbb{R}^{6m \times 6m}$. This ensures the continuity constraint is enforced exactly throughout.

2.6. Algorithm Summary

Algorithm 1 Physics-informed local interpolation at a target $\mathbf{x}^\star$

Require: neighbor indices $\mathcal{N}$, particle data $\{\mathbf{x}_k, \mathbf{u}_k, \mathbf{a}_k\}$, weights choice, parameters $\rho, \mu, \lambda_{\text{vel}}, \lambda_p$

Ensure: $\mathbf{u}(\mathbf{x}^\star)$, $\nabla\mathbf{u}$, $H^{(u_c)}$, and ∇p

Identify m nearest neighbors using spatial search (e.g., k-d tree)

Compute offsets $\Delta\mathbf{x}_k$ and distances r_k

Build weighting vector w_k (normalize)

Construct vector basis $\mathbf{\Phi}$ and assemble $A_{\text{vel}}, b_{\text{vel}}$

Construct momentum block $A_{\text{mom}}, b_{\text{mom}}$ using Eq. (3)

Form weighted system $A = \begin{bmatrix} W_3 A_{\text{vel}} \\ W_3 A_{\text{mom}} \end{bmatrix}$, $b = \begin{bmatrix} W_3 b_{\text{vel}} \\ W_3 b_{\text{mom}} \end{bmatrix}$

Append regularization rows for velocity and pressure subblocks

Append divergence-free constraint for incompressible flows

if using Huber loss **then**

    Solve the initial L_2 problem using Eq. (9) if divergence-free or the corresponding regularized least-squares system otherwise

    Compute MAD from residuals $r^{(0)} = A\theta^{(0)} - b$ and set δ

    for $j = 1$ to $j_{\max}$ **do**

        Update Huber weights $w_i^{(j)}$ using Eq. (11)

        Solve the weighted constrained problem using Eq. (12) if divergence-free, or the corresponding weighted least-squares problem otherwise

        if $\|\theta^{(j+1)} - \theta^{(j)}\| < \text{tol}$ **then**

            break ▷ Converged

        end if

    end for

    $\theta \leftarrow \theta^{(j+1)}$

else

    Solve L_2 problem via KKT system (9) or standard least-squares

end if

Unpack coefficients to produce $\nabla\mathbf{u}$, $H^{(u_c)}$, and ∇p

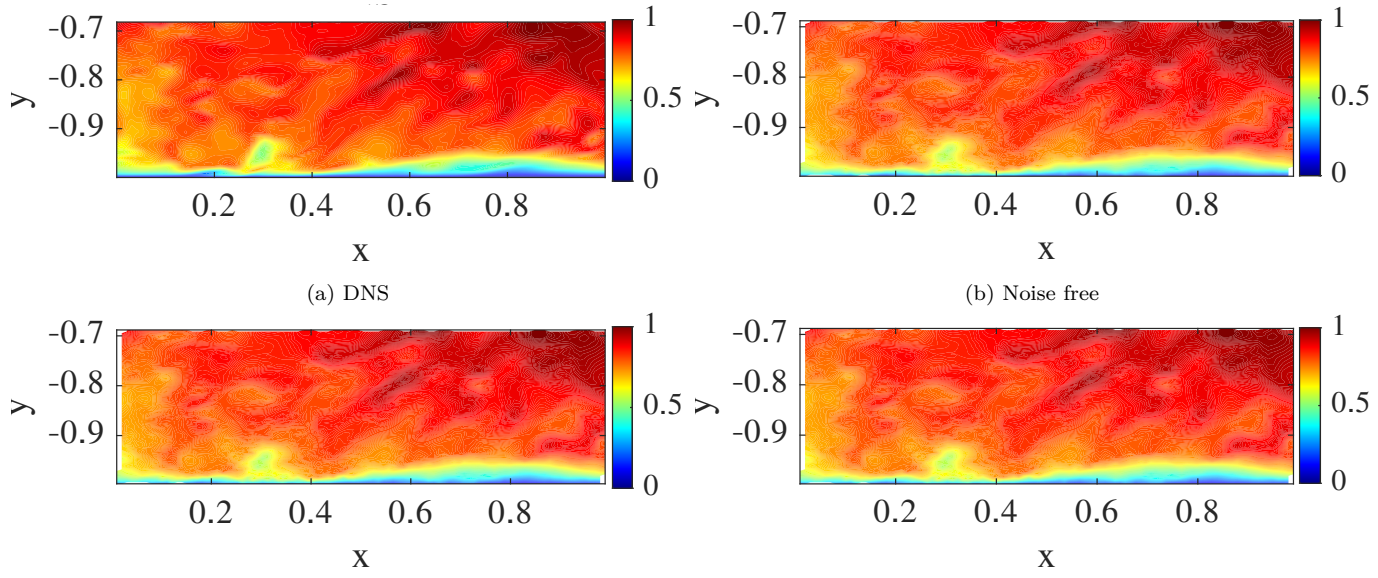
3. Results and Discussions

3.1. DNS channel flow

DNS channel data from JHU turbulence database (Graham et al., 2014) are used in this case. More details can be found in our previous studies (Fenelon et al., 2025). To compare with the ground truth, the data are processed at the same structured grids as the DNS data. A filtering of interpolation locations is performed to ensure that the interpolation is performed within the PTV domain (not on the edges or outside). Locations outside the domain are marked as 'NaN' and are skipped in the processing.

To assess noise sensitivity, artificial positional perturbations are added to all particle locations at four time instances. Assuming 80% of a 2560-pixel camera sensor captures the unit-length flow domain in the streamwise direction, a 0.1-pixel perturbation corresponds to amplitude 4.5×10^{-5}. RBF-QR (Fenelon et al., 2025) is applied to reconstruct particle tracks, from which positions, velocities, and accelerations at the snapshot instants are extracted. Three cases are compared: (1) noise-free; (2) noise added, no acceleration filtering; and (3) noise added, with spatial band-pass acceleration filtering (ACC filtering). All reconstructions use eleven nearest neighbors with both the Huber norm and the divergence-free constraint enabled.

Figure 1 shows the comparisons of interpolated streamwise velocity. The seeding density is 0.0375 ppp, corresponding to 104634 particles. The DNS data has a resolution of $100 \times 100 \times 100$. Both noise-free and noise-added reconstructions agree well with the DNS in flow patterns and amplitudes. Fine-scale details near the bottom wall are slightly under-resolved due to lower local seeding density.



(a) DNS

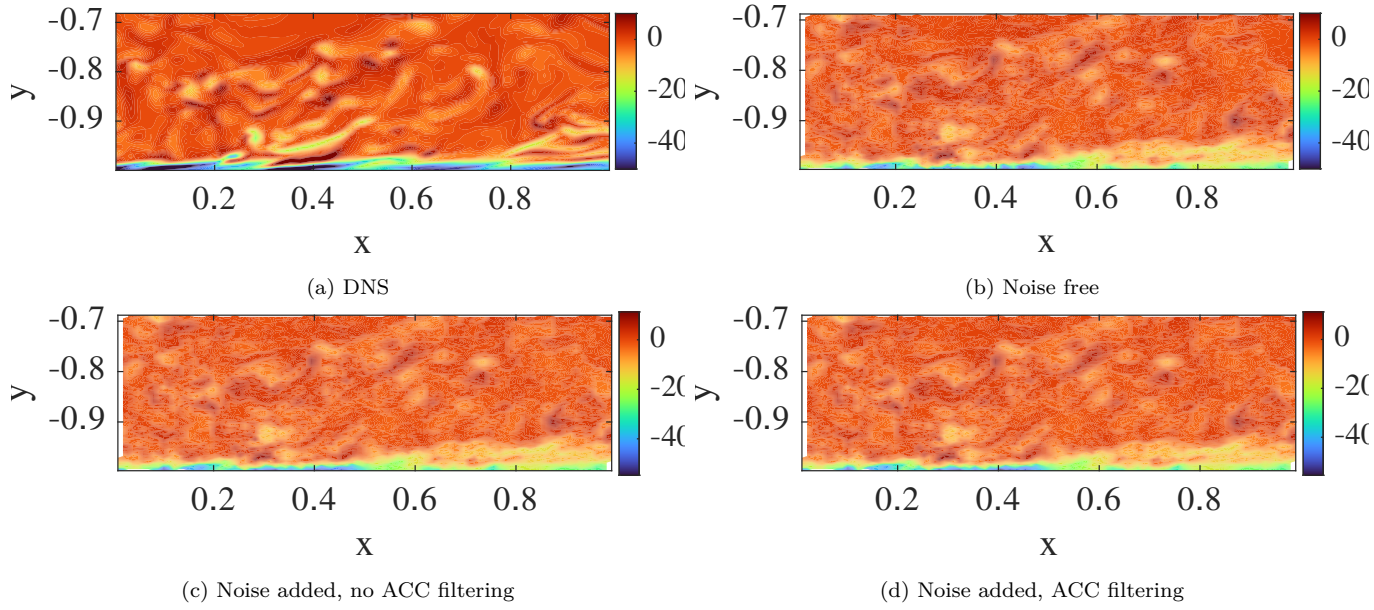
(b) Noise free

(c) Noise added, no ACC filtering

(d) Noise added, ACC filtering

Figure 1. Contour of u-component from a slice cut at $z = 4.6$.

After obtaining the velocity gradients, other flow properties can be derived, such as vorticity and Q-criterion (Q). The spanwise vorticity ω_z interpolated is compared with DNS in Figure 2. The result generally agrees with the ground truth. Some small structures are not well resolved, especially in the region near the wall at $y = -1$. The isosurfaces of $Q = 15$ with surface colored by u are compared in Figure 3. Large-scale coherent structures (hairpin vortices and vortex tubes) are well recovered in the Q isosurfaces. Across all cases, ACC filtering has a negligible effect on velocity reconstructions, confirming the robustness of the kinematic quantities to the noise level considered.



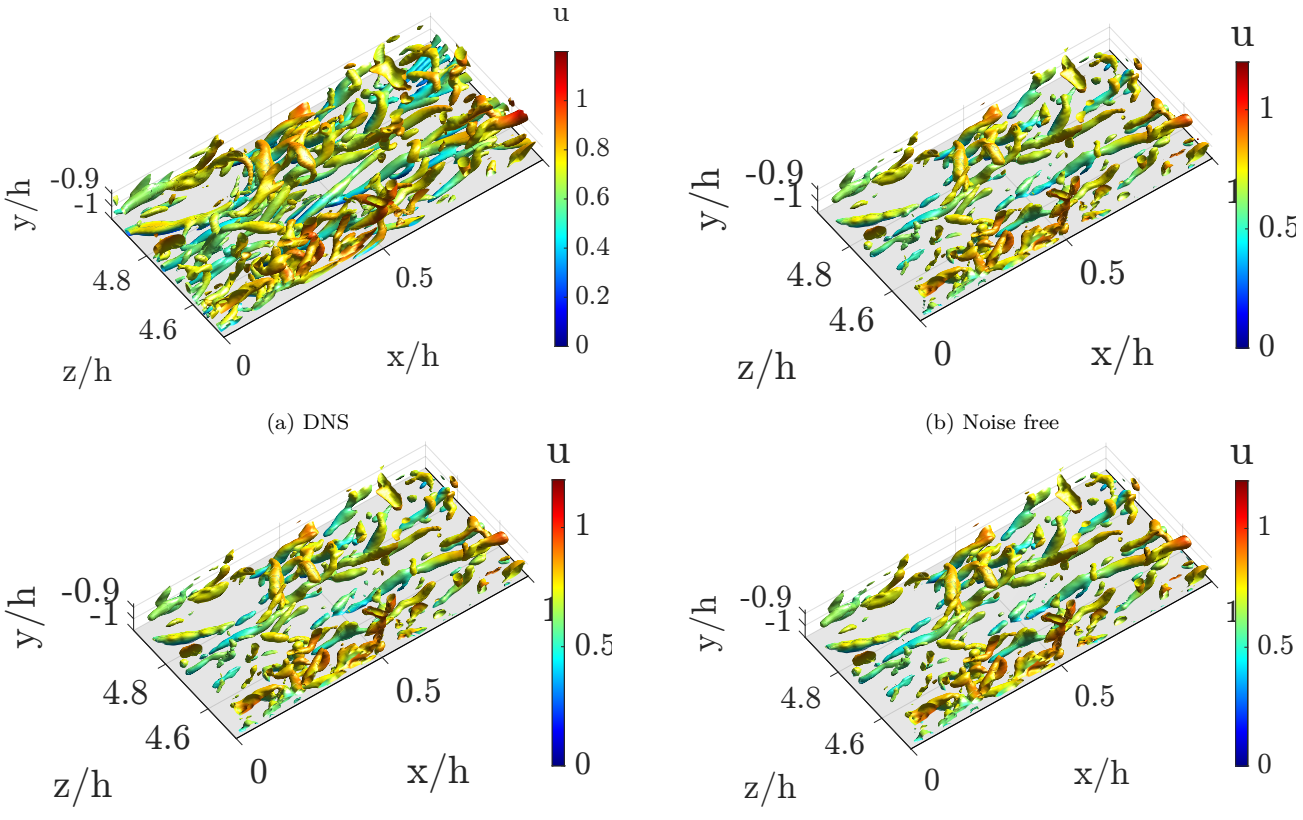
(a) DNS

(b) Noise free

(c) Noise added, no ACC filtering

(d) Noise added, ACC filtering

Figure 2. Contour of nondimensional ω_z from a slice cut at $z = 4.6$.



(a) DNS

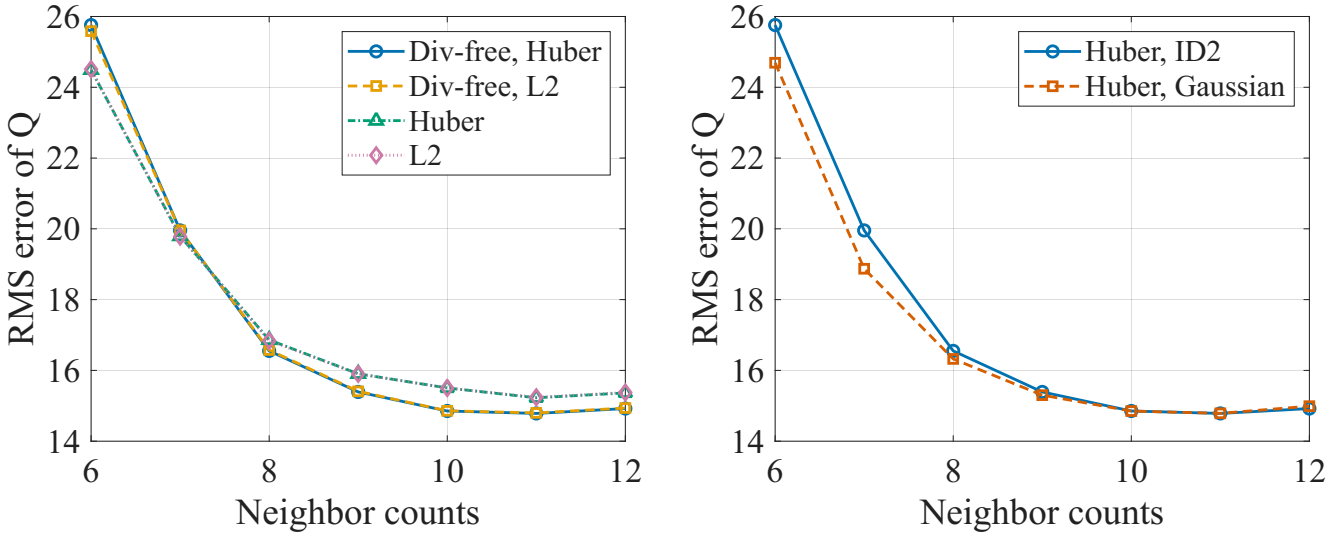
(b) Noise free

(c) Noise added, no ACC filtering

(d) Noise added, ACC filtering

Figure 3. Isosurface of $Q = 15$ (nondimensional) colored by u.

The reconstruction quality is assessed by calculating the root-mean-square (RMS) error of Q, with tuning of the processing parameter over a broad parameter sweep. Figure 4 compares the effects of neighbor count, the divergence-free constraint, spatial weighting kernel, and loss norm. In all configurations, the error is highest at the minimum required neighbor count and decays to a plateau between ten and twelve neighbors. The optimal number of neighbors is expected to vary with seeding density. Figure 4a shows that enabling the divergence-free constraint helps lower the RMS error of Q for neighbor counts greater than seven, while the choice between Huber and $L2$ norms has a comparatively minor effect at this noise level. Figure 4b shows that the ID2 and Gaussian weighting kernels yield nearly identical error curves, indicating that the reconstruction is not strongly sensitive to the specific weighting form once the neighbor count is sufficient.

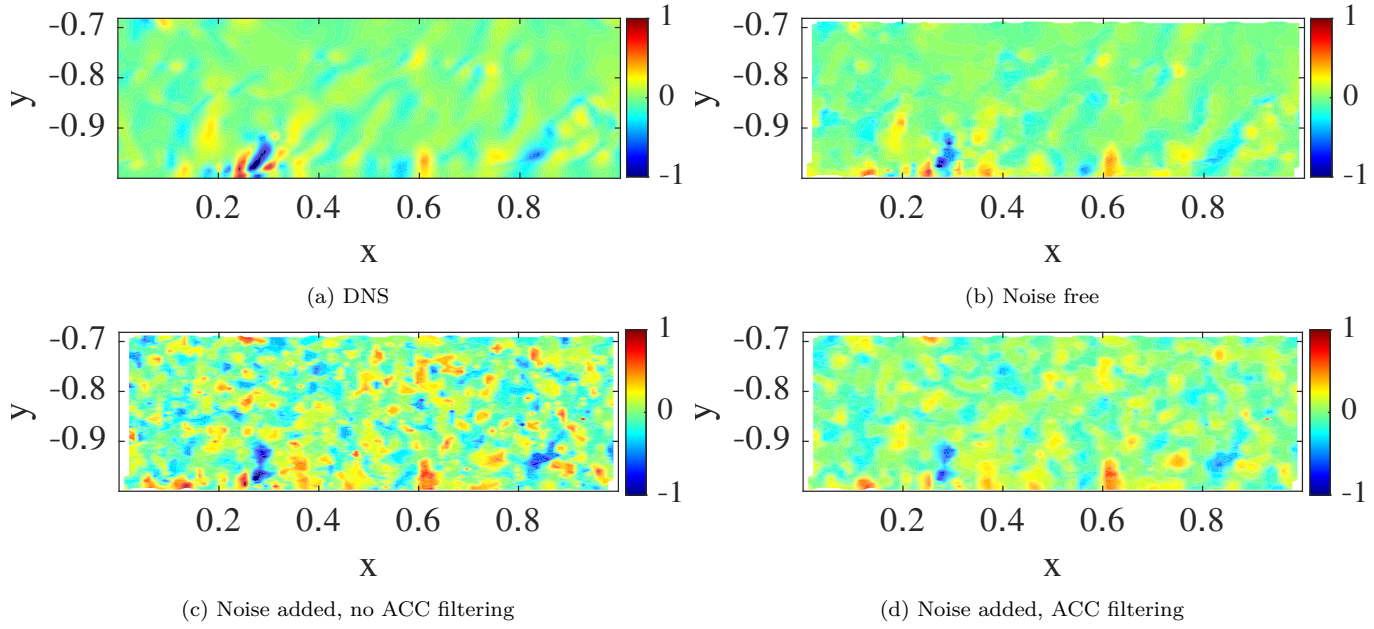


(a) ID2 spatial weighting

(b) Divergence-free

Figure 4. Error metrics for different processing settings. All cases have ACC filtering.

One additional feature of the physics-informed method is the direct calculation of pressure gradients. The figure below compares the results of $\partial p/\partial x$. In the noise-free case, the dominant pressure-gradient structures are well recovered, albeit with slightly underestimated amplitudes. In the case without acceleration filtering, the pressure gradient field is significantly contaminated by noise (Figure 5c). The ACC filtering successfully suppresses the noise. However, it also attenuates the pressure gradient amplitude (Figure 5d).



(a) DNS

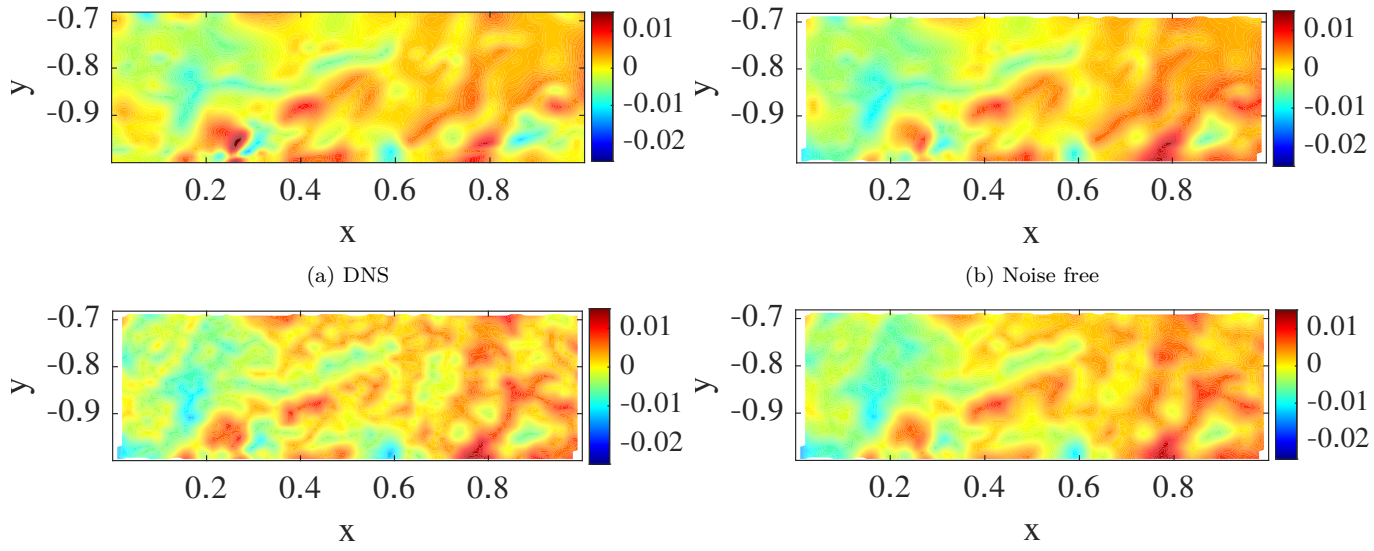
(b) Noise free

(c) Noise added, no ACC filtering

(d) Noise added, ACC filtering

Figure 5. Contour and error map of nondimensional $\partial p/\partial x$ from a slice cut at $z = 4.6$.

In our previous study (Zhang et al., 2025), a pressure solver for MP-PTV was provided. The pressure gradients are calculated at the PTV particle locations, and radial basis function-finite difference (RBF-FD) is used to approximate the gradient operator. The pressure field is solved from an overdetermined linear system and then interpolated using RBF at the structured grids. In the current approach, the pressure gradients are already obtained at structured grids, which allows us to apply existing PIV-pressure solvers directly. One-Shot Matrix-based Omnidirectional Integration Solver (OS-MODI) (Zigunov & Charonko, 2024) is applied to the pressure gradients calculated. This method subtracts the mean value from the domain, giving a differential pressure field. A Dirichlet boundary condition, a reference pressure node, can be used to correct the pressure value. The pressure field is calculated and compared with the DNS result in Figure 6. The noise-free case closely reproduces the DNS pressure field. Although the case with no ACC filtering shows some scattered noise, the major flow pattern is well reconstructed. With ACC filtering, the reconstructed pressure field shows good quantitative agreement with the ground truth.



(a) DNS

(b) Noise free

(c) Noise added, no ACC filtering

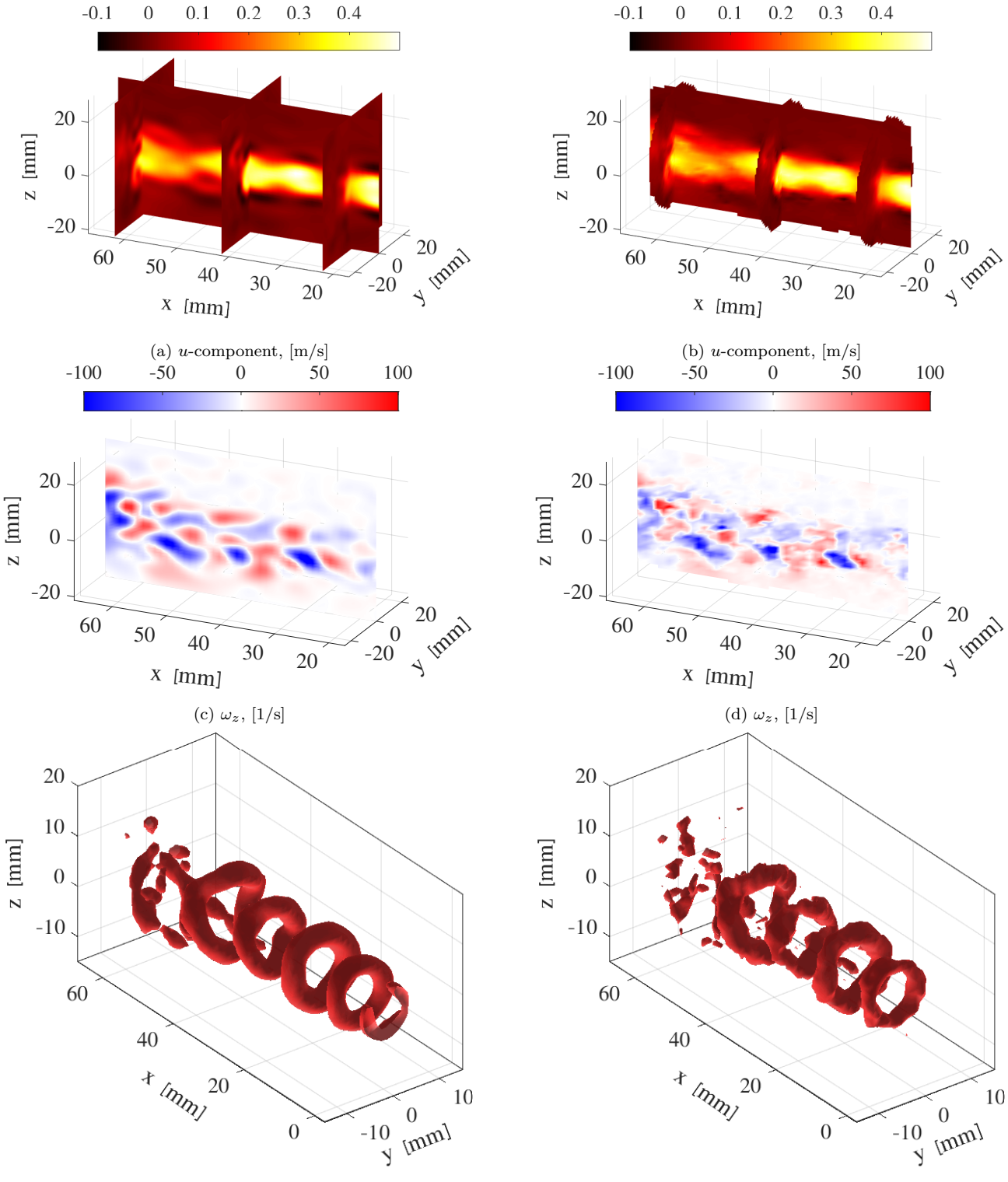
(d) Noise added, ACC filtering

Figure 6. Slice of nondimensional pressure field at $z = 4.6$ with global mean subtracted.

3.2. Experimental results

The experimental dataset is a time-resolved (1300 Hz) PTV measurement of a turbulent water jet, available as a LaVision DaVis example project (LaVision GmbH, 2026). Particle tracks are processed using the Shake-The-Box algorithm in DaVis 10.2, and only the particle locations of each track are used herein. First, the pathlines are approximated using RBF-QR (Fenelon et al., 2025) at four successive time instants, which also provides velocities and accelerations at the snapshot time instants. Results of 200 PTV snapshots are mapped to structured grids using the current algorithm (11 neighbors, ID2 weighting, Huber norm, divergence-free constraint). For comparison, VIC# (Jeon et al., 2022) reconstructions at coincident grid points are used as reference.

Instantaneous flow fields visualized via streamwise velocity, spanwise vorticity, and swirling-strength isosurfaces (Figure 7) show a high level of agreement between the current method and VIC# in position, magnitude, and coherent structures. Vortex rings concentrated along the jet shear layers are identified by both methods with closely matched magnitudes. Although the gradient calculation methods are different, vorticity fields are broadly consistent.



(a) u-component, [m/s]

(b) u-component, [m/s]

(c) ω_z, [1/s]

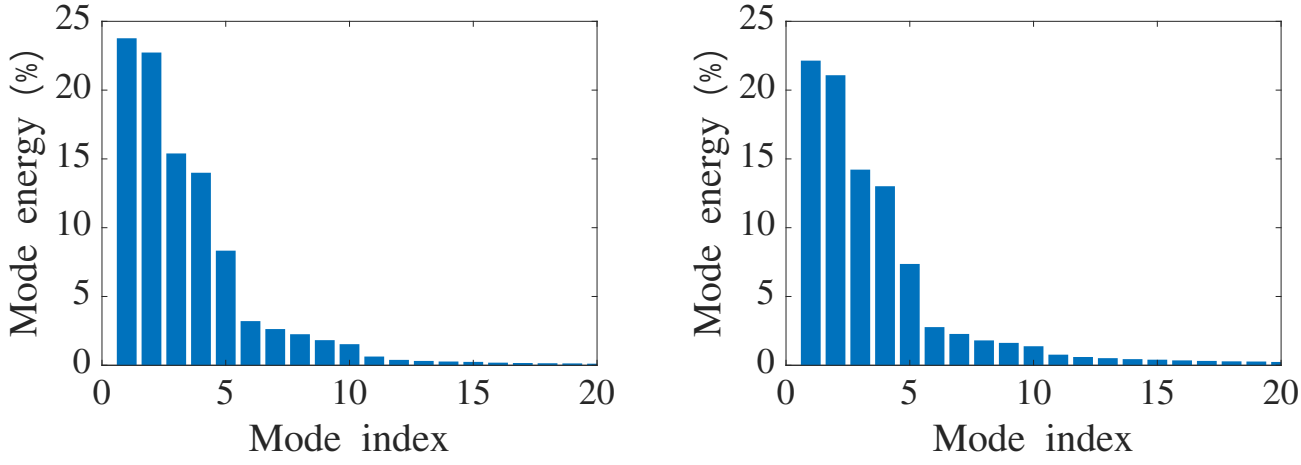
(d) ω_z, [1/s]

(e) Swirling strength, [1/s^2]

(f) Swirling strength, [1/s^2]

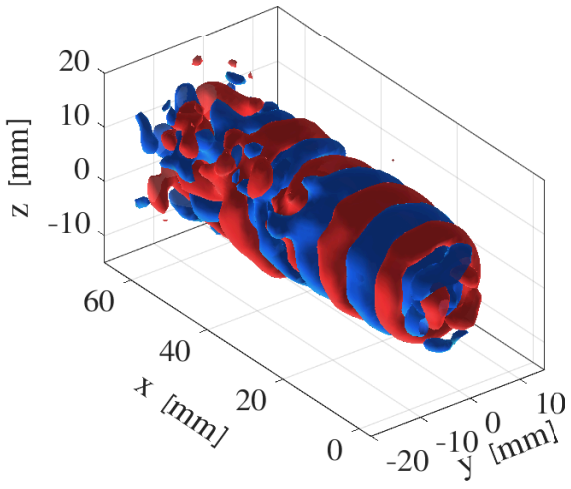
Figure 7. Comparison of instantaneous flow field. Isosurface value of swirling strength is 2500. Left column: VIC# results; right column: current method results.

Then Proper Orthogonal Decomposition (POD) (Lumley, 1967) is applied to the 200 3D velocity fields using the method of snapshots (Sirovich, 1987). The modal energy distributions are similar between the two methods, though VIC# concentrates slightly higher energy in the leading modes due to its global smoothing. The leading POD mode shapes (Figure 9 and Figure 10) from both methods identify the same dominant coherent structures near the jet exit—organized rings or helical structures breaking into smaller scales downstream—with axial symmetry in the u component and planar symmetry in v and w. These are consistent with the structure of axisymmetric jet instabilities. Mode 3 captures flow structures at smaller spatial scales. VIC# results are noticeably smoother, and the present method retains more fine-scale variability, which is consistent with its higher sensitivity to local particle distribution rather than a global regularization. Spatial correlation coefficients between corresponding POD modes from the two methods range from 0.92–0.97 for modes 1–5, confirming strong agreement in dominant modal content. Per-snapshot computational cost of the current method is substantially lower than VIC# for the cases examined.

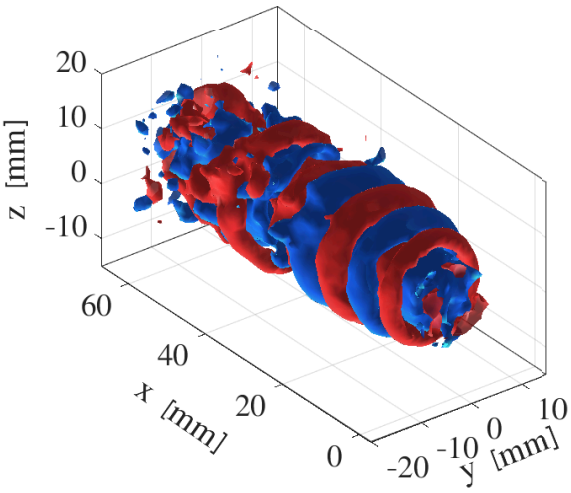


(a) VIC# (b) Current method

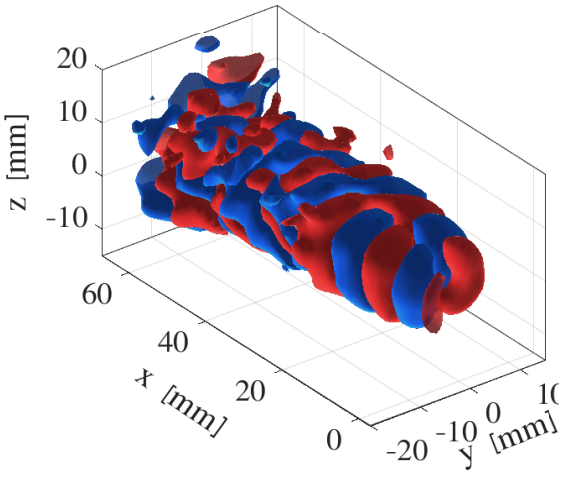
Figure 8. Energy distribution of POD modes.



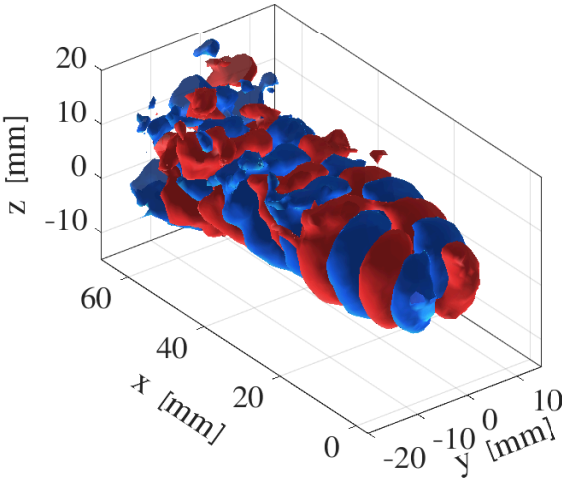
(a) u-component, mode 1



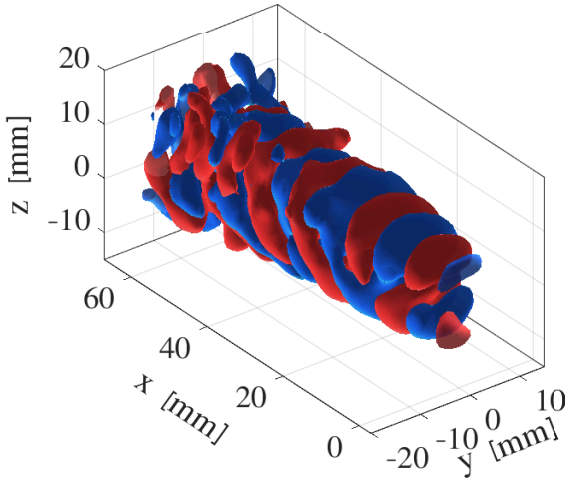
(b) u-component, mode 1



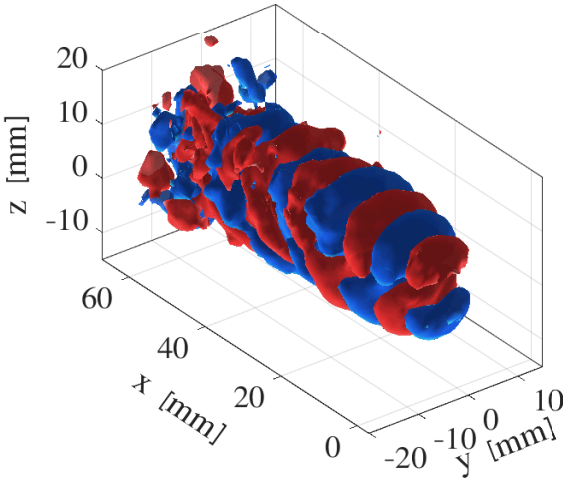
(c) v-component, mode 1



(d) v-component, mode 1

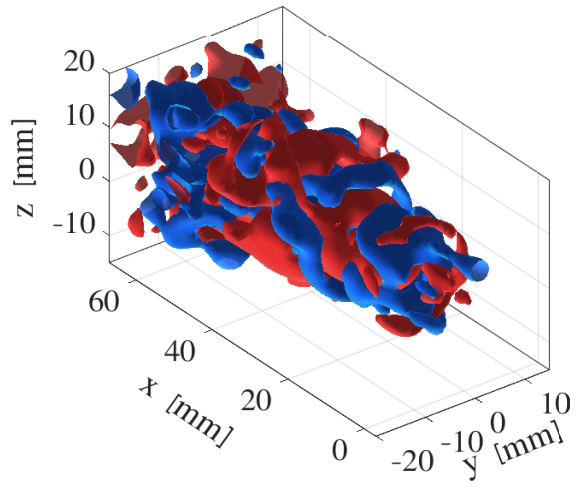


(e) w-component, mode 1

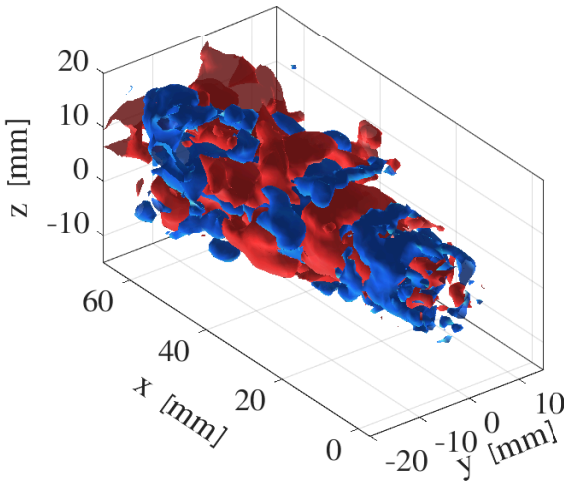


(f) w-component, mode 1

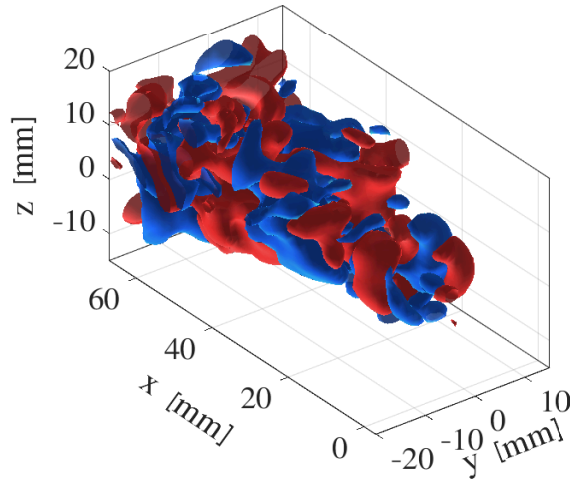
Figure 9. Comparison of the most dominant POD modal shapes. Isosurface value of ±0.1. Left column: VIC# results; right column: current method results.



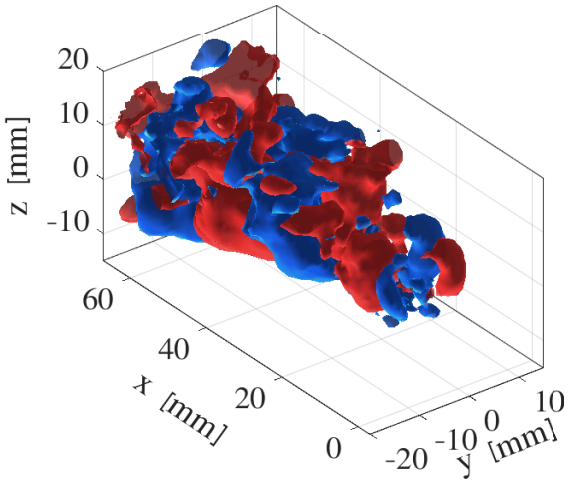
(a) u-component, mode 3



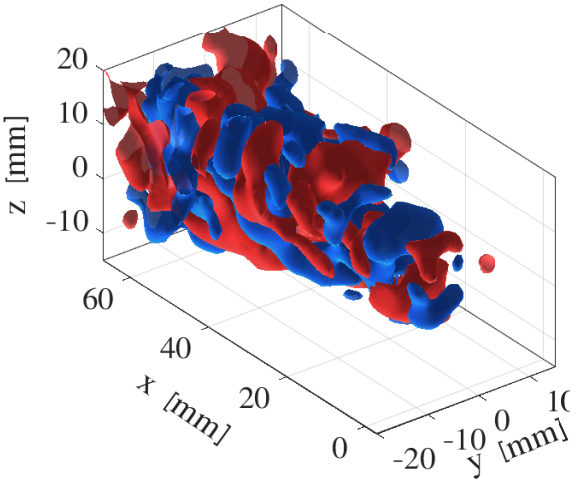
(b) u-component, mode 3



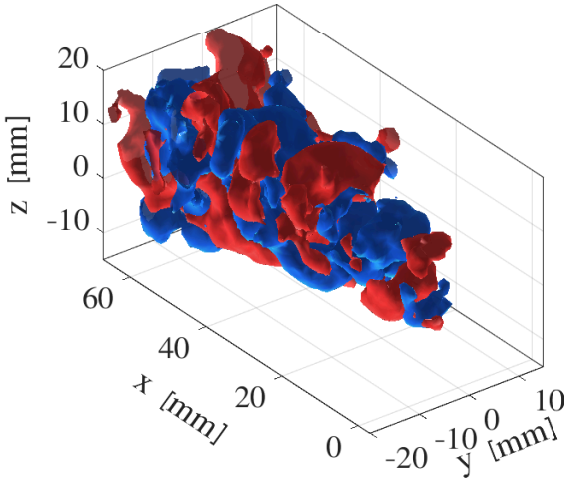
(c) v-component, mode 3



(d) v-component, mode 3

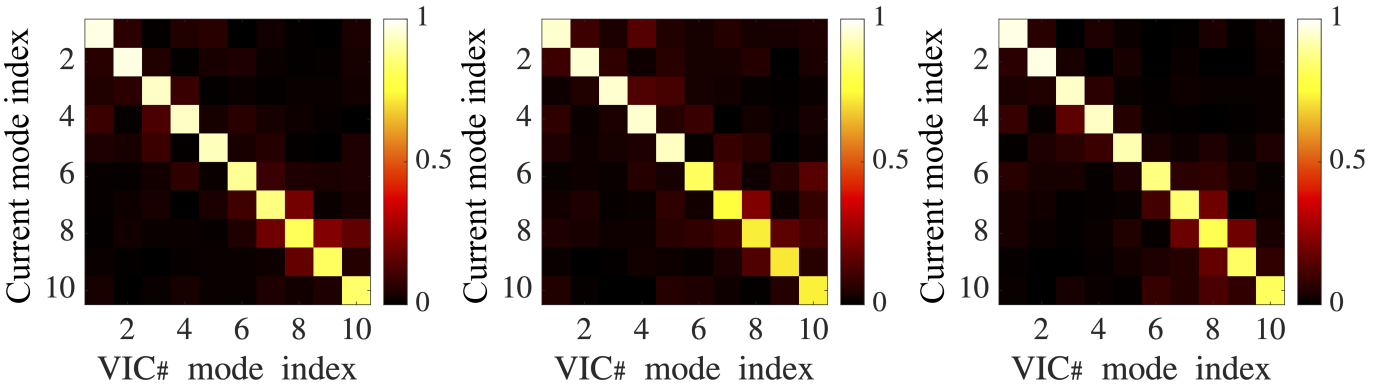


(e) w-component, mode 3



(f) w-component, mode 3

Figure 10. Comparison of the third dominant POD modal shapes. Isosurface value of ±0.1. Left column: VIC# results; right column: current method results.



(a) u-component (b) v-component (c) w-component

Figure 11. Spatial correlation coefficient of POD modes from two methods.

4. Conclusions

We developed a physics-informed local interpolation framework that reconstructs velocity, gradients, Hessians, and pressure gradients from scattered PTV measurements. By embedding the Navier–Stokes momentum equation within a weighted least-squares system, the method ensures that interpolated fields remain dynamically consistent with measured material accelerations. Validation using DNS channel-flow data confirms accurate recovery of velocity and vorticity fields as well as pressure gradient distributions, including large-scale coherent structures and near-wall features, under both noise-free and noise-corrupted conditions. Acceleration filtering reduces pressure-gradient RMS error and integrated pressure fields obtained via OS-MODI show good quantitative agreement with DNS. Application to experimental PTV measurements of a turbulent water jet demonstrates robustness to real data, enabling 3D field reconstruction and subsequent modal analysis. Agreement with VIC# in dominant structures, modal energy content, and leading POD modes—at substantially lower per-snapshot computational cost—indicates that the current method provides a practically competitive alternative for routine post-processing of large-scale 3D PTV datasets.

The proposed framework provides a unified bridge between Lagrangian particle-tracking measurements and Eulerian grid-based flow diagnostics. Its ability to directly compute pressure gradients on structured grids allows seamless integration with existing PIV-based pressure solvers, reducing reliance on intermediate interpolation stages. Future work will address more systematic uncertainty quantification, adaptive neighbor selection, and near-wall treatment.

Acknowledgements

We gratefully acknowledge support from ONR (Grant Number N00014-24-12516), monitored by Dr. Leighton Meyers.

References

Fenelon, M. R., Zhang, Y., Schmid, P. J., & Cattafesta, L. N. (2025, May). Kinematic decomposition of volumetric particle tracking velocimetry data. *Experiments in Fluids*, *66*(6). doi: 10.1007/s00348-025-04035-8

Fornberg, B., & Flyer, N. (2015). *A primer on radial basis functions with applications to the geosciences*. Society for Industrial and Applied Mathematics. doi: 10.1137/1.9781611974041

Gesemann, S., Huhn, F., Schanz, D., & Schröder, A. (2016). From noisy particle tracks to velocity, acceleration and pressure fields using b-splines and penalties. In *18th international symposium on applications of laser techniques to fluid mechanics* (pp. 1–17). (Paper 04.5_4_186)

Ghosh, S., Mons, V., Sipp, D., & Schmid, P. J. (2024). A robust computational framework for variational data assimilation of mean flows with sparse measurements corrupted by strong outliers. *Journal of Computational Physics*, *508*, 113008.

Graham, J., Lee, M., Malaya, N., Moser, R., Eyink, G., & Meneveau, C. (2014). *Turbulent channel flow data set* (Tech. Rep.). Johns Hopkins University.

Jeon, Y. J., Müller, M., & Michaelis, D. (2022, April). Fine scale reconstruction (vic#) by implementing additional constraints and coarse-grid approximation into vic+. *Experiments in Fluids*, *63*(4). doi: 10.1007/s00348-022-03422-9

LaVision GmbH. (2026). *DaVis Example Projects.* `https://www.lavision.de/en/downloads/software/davis-example-projects.php`. (Accessed: 2026-05-28)

Lumley, J. L. (1967). The structure of inhomogeneous turbulent flows. In A. M. Yaglom & V. I. Tatarsky (Eds.), *Atmospheric turbulence and radio wave propagation* (pp. 166–178). Moscow: Nauka.

Raissi, M., Perdikaris, P., & Karniadakis, G. (2019, February). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, *378*, 686–707. doi: 10.1016/j.jcp.2018.10.045

Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of Applied Mathematics*, *45*(3), 561–571.

Zhang, Y., Cattafesta, L., & Schmid, P. (2025). Estimation of velocity gradient and hessian tensors from particle tracking velocimetry. In *16th international symposium on particle image velocimetry – ispiv 2025 june 26–28.* Tokyo, Japan.

Zigunov, F., & Charonko, J. J. (2024, March). A fast, matrix-based method to perform omnidirectional pressure integration. *Measurement Science and Technology*, *35*(6), 065302. doi: 10.1088/1361-6501/ad2da5