Available online at https://link.springer.com/journal/42241
http://www.jhydrodynamics.com
Journal of Hydrodynamics, 2025
https://doi.org/10.1007/s42241-025-0062-x

Enhancing turbulent channel flow super-resolution via physics-guided deep learning with global pressure correlations

Mao-kun Ye, Guo-qing Fan, De-cheng Wan*
Computational Marine Hydrodynamics Lab (CMHL), School of Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

(Received July 23, 2025, Revised August 15, 2025, Accepted August 16, 2025, Published online November 27, 2025)

Abstract: High-fidelity simulation of incompressible turbulent channel flows at high Reynolds numbers remains computationally intensive due to their multi-scale nature, necessitating efficient strategies to reconstruct high-resolution (HR) fields from low-resolution (LR) data. This study introduces a physics-guided neural network (PGNN) framework that leverages the elliptic nature of the pressure field in incompressible flows to enhance super-resolution (SR) reconstruction of turbulent velocity fields. Inspired by the global correlation-capturing capabilities of attention mechanisms, we incorporate LR pressure data as auxiliary input features to guide neural networks in inferring local velocity components, thereby explicitly encoding physical constraints into the learning process. High-fidelity training datasets are generated via large-eddy simulations, with LR fields derived through spatial filtering using down-sampling ratios (DSR) of 4, 8. A physics-guided U-Net (pgU-Net) is developed, contrasting with a pure data-driven U-Net baseline. Results demonstrate that integrating pressure fields significantly improves reconstruction accuracy, particularly under challenging DSR of 8, the pgU-Net achieves an R^2 score of 0.8048, outperforming the data-driven model (0.6792) and bi-cubic interpolation (–0.1801). By leveraging pressure fields as physical knowledge encoding global flow correlations, our framework effectively addresses the multi-scale reconstruction challenges in turbulent flows while maintaining computational efficiency. By embedding elliptic pressure correlations as physical priors, this framework pioneers a hybrid approach that bridges data-driven learning and fluid physics, offering a robust approach to reduce computational costs while enhancing the fidelity of turbulent flow predictions.

Key words: Super-resolution reconstruction, machine learning, turbulent flow, physics-guided

0. Introduction

Turbulent flow modeling stands as a persistent and fundamental challenge in the realm of fluid dynamics. This challenge is particularly pronounced in high Reynolds number (*Re*) applications, where direct numerical simulation (DNS) becomes computationally intractable due to the vast computational resources required to resolve the intricate details of the flow. Although large eddy simulation offers a practical alternative by resolving the large-scale motions and modeling the subgrid-scale turbulence, it still demands substantial computational resources when applied to complex geometries[1-2] or at industrial scales. In recent years, the rapid development of machine learning has presented new opportunities for predicting turbulent flows. Among these, super-resolution reconstruction techniques have emerged as a promising approach, enabling the recovery of high-resolution flow fields from coarse numerical solutions.

Early data-driven super-resolution (SR) methods treated flow fields as generic images, employing convolutional neural networks (CNNs) to establish direct mappings between low-resolution (LR) and high-resolution (HR) velocity fields. Fukami et al.[3] had been the first to introduce this paradigm, developing a CNN-based framework for reconstructing two-dimensional turbulent wakes and homogeneous turbulence. Subsequently, they extended this framework to spatio-temporal SR[4-5]. Liu et al.[6] enhanced the prediction accuracy by incorporating temporal dependencies through multi-path networks. Bi et al.[7] integrated attention mechanisms into SRCNN architectures, which enabled the models to capture global spatial correlations and led to significant performance improvements. More recently, transformer-

Project supported by the National Natural Science Foundation of China (Grant No. 52131102).
Biography: Mao-kun Ye (1994-), Male, Ph. D.,
E-mail: maokunye@sjtu.edu.cn
Corresponding author: De-cheng Wan,
E-mail: dcwan@sjtu.edu.cn

based models[8] and hybrid attention frameworks[9] have leveraged multi-scale feature fusion and long-range dependency modeling, demonstrating superior reconstruction capabilities for both isotropic and anisotropic turbulent properties. Sofos et al.[10] proposed a lightweight U-Net variant for spatiotemporal forecasting of high-speed flows, achieving a balance between accuracy and computational efficiency. At the core of these studies is the training of neural network models to take low-resolution flow fields as input and output high-resolution results. This approach allows high-resolution flow fields to be obtained by conducting computational fluid dynamics (CFD) simulations on coarser grids, thereby significantly reducing the computational resource requirements. Notably, these studies predominantly adopted pure data-driven frameworks. That is, during the training of neural network models, the "flow field" was treated merely as an "image", and the super-resolution of the flow field was more or less considered a pure image-processing problem. As a result, limited or minimal physical knowledge of the flow field was incorporated into the modeling process.

In recent years, there has been an increasing research focus on integrating more prior physical knowledge into the modeling of artificial neural networks[11-12]. This approach aims to enhance the accuracy, interpretability, and generalizability of neural networks by directly embedding fundamental physical principles into the learning process. Specifically, the incorporation of physical knowledge into neural networks can be achieved at either the output or the input end of the model. The first approach, known as physics-informed neural networks (PINNs)[13-15], introduces physical constraints directly into the optimization process of the neural network, ensuring that the model's predictions comply with fundamental physical laws[16]. The second approach, commonly referred to as physics-guided neural networks (PGNNs)[17-18], involves feeding the neural network with lower-fidelity data as a guiding framework during training. This method utilizes approximate physical models or data to steer the learning process, enhancing the model's generalization ability and enabling it to produce physically consistent results.

Inspired by the above research works, in this study, we propose a data-driven modeling framework based on the concept of PGNNs for the super-resolution reconstruction of turbulent channel flows from low-resolution flow fields. Our approach capitalizes on the elliptic nature of the pressure equation in incompressible flows to improve the performance of the neural network. Drawing inspiration from the attention mechanism, which is widely used in natural language processing and time series analysis to capture global correlations, we incorporate pressure data from globally distributed sensors as additional input features. This enables the model to infer local flow quantities, specifically the three instantaneous velocity components, while naturally accounting for global correlations across different spatial points. By integrating these strategies, our framework effectively combines physical principles with data-driven learning, thereby enhancing the accuracy and robustness of super-resolution reconstruction in turbulent flows.

To demonstrate the effectiveness of incorporating pressure data, we first designed a benchmarking case. In this case, we directly connected the low-resolution U field and the high-resolution U field using a U-Net network. Given that the U-Net network is already an extremely efficient multi-scale spatial feature extractor capable of effectively modeling the spatial information of the same channel[19], we use it as the baseline in this study. In contrast, our proposed pgU-Net incorporates the pressure field as an additional input to the U-Net network. By leveraging the U-Net network's efficient multi-scale and inter-channel feature extraction and modeling capabilities, we integrate the global information of the flow field embedded in the pressure field into the neural network model.

The remainder of this paper is structured as follows. First, we provide a detailed description of the generation process of the high-fidelity dataset. Next, we introduce the neural network (NN) models employed in this study, namely the U-Net and pgU-Net. This includes an in-depth exploration of their architectures, the mapping relationships they aim to establish, and the preparation of the training dataset. Subsequently, the NN models are trained and applied to the super-resolution reconstruction of fully developed turbulent flows, with lower-resolution data serving as the input. Finally, based on the results and discussions presented, we draw comprehensive conclusions.

1. Generation of high-fidelity dataset

In machine learning tasks, a high-quality dataset is considered a fundamental prerequisite. Therefore, in this chapter, we will detail the process of generating the high-quality dataset used in this study.

1.1 *CFD Approach*

After applying a spatial filter to the incompressible Navier-Stokes equations, the governing equations can be written as:

$$\frac{\partial \tilde{u}_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial \tilde{u}_i}{\partial t}+\frac{\partial \tilde{u}_i \partial \tilde{u}_j}{\partial x_j}=-\frac{1}{\rho}\frac{\partial \tilde{p}}{\partial x_i}+\nu\frac{\partial^2 \tilde{u}_i}{\partial x_j \partial x_j}-\frac{\partial \tau_{ij}}{\partial x_j} \tag{2}$$

where $\tilde{u}_i\,(i=1,2,3)$ is the filtered velocity component in the x_i direction. p is the filtered pressure, ν the kinematic viscosity of the fluid and τ_{ij} is the so-called sub-grid stress which is given by:

$$\tau_{ij}=\frac{2}{3}k_{\mathrm{SGS}}\delta_{ij}-2\nu_{\mathrm{SGS}}\tilde{S}_{ij} \tag{3}$$

where $\tilde{S}_{ij}$ is the resolved strain-rate tensor, while ν_{SGS} being the unresolved eddy viscosity and need additional model to close. In the current work, the wall-adapting local eddy-viscosity (WALE) model[20-22] is applied, and the unresolved eddy viscosity is written as:

$$\nu_{\mathrm{SGS}}=(C_w\Delta)^2\frac{(S_{ij}^d S_{ij}^d)^{3/2}}{(\tilde{S}_{ij}\tilde{S}_{ij})^{5/2}+(S_{ij}^d S_{ij}^d)^{5/4}} \tag{4}$$

where C_w is the WALE coefficient, Δ is the cube root of local cell volume and S_{ij}^d is the traceless symmetric part of the square of the velocity gradient tensor which is defined by:

$$S_{ij}^d=\frac{1}{2}\left(\frac{\partial \tilde{u}_i \partial \tilde{u}_k}{\partial x_k \partial x_j}+\frac{\partial \tilde{u}_j \partial \tilde{u}_k}{\partial x_k \partial x_i}\right)-\frac{1}{3}\delta_{ij}\frac{\partial \tilde{u}_k \partial \tilde{u}_l}{\partial x_l \partial x_k} \tag{5}$$

1.2 *Numerical setup*

Figure 1 illustrates the computational domain used in the CFD simulations for this study. The domain dimensions are $L_x\times L_y\times L_z=2\pi\delta\times 2\delta\times\pi\delta$, where δ represents the channel half-height. The coordinate system is defined as follows: x corresponds to the streamwise direction, y to the wall-normal direction and z to the spanwise direction. Periodic boundary conditions are applied to the side walls, while the top and bottom walls are set as no-slip boundaries.

In this study, the Reynolds numbers are defined based on the bulk velocity and friction velocity. Specifically, the bulk Reynolds number is given by $Re_{\mathrm{b}}=U_{\mathrm{b}}\delta/\nu=20\,000$, and the friction Reynolds number is $Re_\tau=u_\tau\delta/\nu=1\,000$. Here, $u_\tau=\sqrt{\tau_w/\rho}$ represents the friction velocity at the wall, where τ_w is the wall shear stress, ρ is the fluid density.

The specifications of the computational grid are detailed in Table 1. The grid consists of 320 cells in the streamwise direction (x), 216 cells in the wall-normal direction (y) and 320 cells in the spanwise direction (z). This configuration results in a total of approximately 2.2×10^7 grid points used in the CFD simulations.

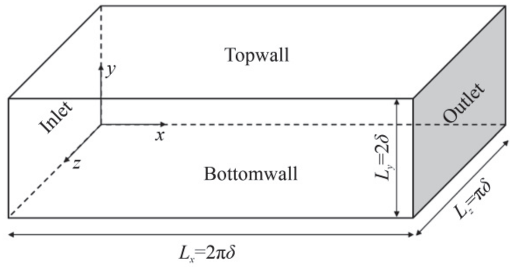


Fig. 1 Illustration of the computational domain

Table 1 Computational grid specifications

Parameter	Value
Δx^+	19.6
Δy^+	0.79-15.0
Δz^+	9.8
$N_x\times N_y\times N_z$	320×216×320
N_{total}	2.21×10^7

1.3 *Numerical results*

The verification procedures for the CFD results were thoroughly addressed in prior work by the authors and are excluded here to maintain conciseness. In the present study, we focus on analysing flow field snapshots derived from the LES database. Figure 2 illustrates instantaneous vortex patterns identified through the Q-criterion, which highlights coherent turbulent structures.

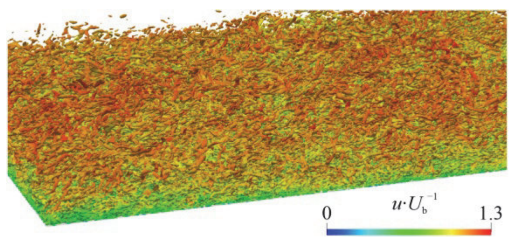


Fig. 2 LES results

2. Definition of NN models for the reconstruction of turbulent channel flow

2.1 *Definition of mapping relation*

For the incompressible Navier-Stokes equations, the pressure solution at an arbitrary point x_0 within the flow field can be expressed as follows:

$$p(x_0) = -\frac{\rho}{4\pi}\iiint_V \frac{\partial u_i}{\partial x_j}\frac{\partial u_j}{\partial x_i}\frac{1}{|x - x_0|}\mathrm{d}x \tag{6}$$

It should be noted that the pressure at any given point in the flow field is a composite result of the influences exerted by all other points. In other words, conversely, the global spatial correlation of the flow field is inherently embedded in the obtained pressure field.

In the realm of modern artificial intelligence, global correlation modeling is often achieved through attention mechanisms, such as self-attention in spatial attention. However, these mechanisms usually impose an additional computational burden during the training process. In contrast, the pressure field can capture global correlations without incurring such extra costs. Therefore, the elliptic nature of the pressure field naturally encodes global flow correlations, providing an ideal and computationally efficient method to model these correlations in turbulent flows.

In light of this, in this study, the mapping relationships established by the neural network (NN) model are defined in Eqs. (6), (7). Equation (6) functions as a benchmark for comparative analysis. It offers a reference basis for evaluating the performance of the proposed PGNN models, which is represented by Eq. (7). In the pgU-Net model, we directly incorporate the low-resolution pressure field as an additional input channel. By doing so, we can perform physically-guided modeling of the global correlations in the turbulent flow field. This approach enables the model to better capture the complex relationships within the flow field and improve the accuracy of super-resolution reconstruction.

$$(U^{\mathrm{HR}}) = \mathrm{U\text{-}Net}(U^{\mathrm{LR}}, U^{\mathrm{LR}}) \tag{7}$$

$$(U^{\mathrm{HR}}) = \mathrm{pgU\text{-}Net}(U^{\mathrm{LR}}, p^{\mathrm{LR}}) \tag{8}$$

Furthermore, to control variables related to the learning efficiency of the neural network as much as possible, we take specific measures regarding the input channels. Although, as defined by Eq. (7), in this study, we input two identical channels, namely U^{LR} and U^{LR}. This is to ensure that the number of input channels in the U-Net network is consistent with that of the pgUNet. By maintaining the same number of input channels, we can more fairly compare the performance of the two models. This way, when evaluating the impact of adding the pressure field as an input in the pgU-Net, we can isolate this factor more effectively and accurately assess its contribution to the model's performance, without being confounded by differences in the number of input channels.

2.2 *Definition of NN models*

The U-Net is a well-known deep-learning architecture that was initially developed for the specific task of medical image segmentation. Its design is characterized by an efficient symmetric encoder-decoder structure, which, along with its remarkable feature fusion capabilities, has enabled its widespread application in various fields beyond medical imaging. In particular, it has found great utility in modeling mapping relationships for two-dimensional data, such as images. The U-Net's core design follows a U-shaped symmetry. The encoder part of the network plays a crucial role in extracting global features from the input image. It does this through a series of convolutional and pooling operations. These operations help in reducing the spatial dimensions of the data while simultaneously capturing important information at different scales. On the other hand, the decoder is responsible for reconstructing the output at the original resolution. It achieves this by using up-sampling convolutional layres to increase the spatial dimensions and, more importantly, by leveraging skip connections. These skip connections enable the decoder to fuse the shallow-level details from the earlier layers of the encoder with the deep-level semantic information obtained from the later layers of the encoder. This fusion process is essential for achieving accurate pixel-level localization in the output. Figure 3 provides an illustration of the U-Net neural network employed in this paper, which clearly shows the architecture and the flow of data through the encoder, decoder, and skip connections.

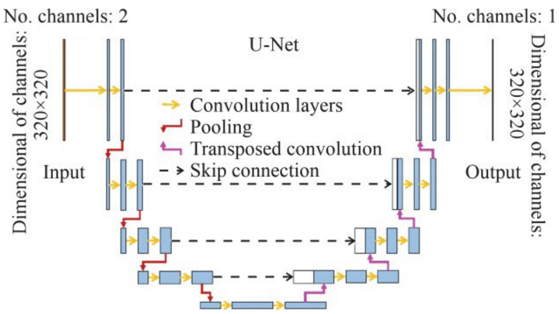


Fig. 3 Illustration of the U-Net structure

2.3 *Preparation of training dataset*

The flow field illustrated in Fig. 2 serves as the basis for generating the data used in training the neural network models. Initially, this high-resolution (HR) flow field is down-sampled to form low-resolution (LR) data. The resulting LR data is then utilized as the training data for the two neural network (NN) models in this study. Specifically, two different down-sampling ratios (DSR) are applied to derive the

LR data from the HR data. These ratios are set to 4, 8. By down-sampling with a ratio of 4, the dimensions of each snapshot in the resulting LR dataset change. If the original HR snapshot had dimensions that could be considered as a larger matrix, after down-sampling with a DSR of 4, the dimensions of a single snapshot become 80×80. Similarly, when the down-sampling ratio is 8, the dimensions of a single snapshot in the LR dataset are 40×40. In both cases, the total number of snapshots in the dataset is 1 000. This number represents a set of different instances or time-steps of the flow field. Figure 4 shows the down-sampled velocity and pressure fields at a specific time instance when the DSR is 8, providing a visual representation of how the data looks after down-sampling.

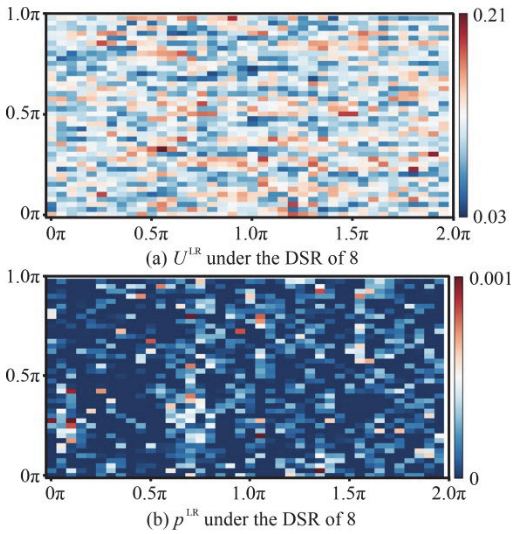


Fig. 4 LR flow field under the DSR of 8

In this paper, to ensure compatibility with the input requirements of the U-Net neural network, an upscaling operation is carried out first on the down-sampled LR data. The bi-cubic interpolation method is employed for this purpose. This method is chosen because it can effectively increase the resolution of the LR data while maintaining a certain level of image quality. Through this upscaling, the dimensions of each LR snapshot, which were either 40×40 or 80×80 after down-sampling, are increased back to 320×320. These up-scaled snapshots are then used as the input for the NN models. After upscaling, a min-max normalization step is applied to the data. This normalization process is essential as it standardizes the original data by scaling all input features to the range $[0,1]$. By doing so, it helps to enhance the training efficiency of the neural network. The min-max scaler is defined as:

$$x_{\text{normalized}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{9}$$

Further, the snapshots in the LR dataset are divided into three subsets: a training dataset, a validation dataset, and a testing dataset. The division is done in a ratio of 7:2:1. The data points in the training dataset are used to train the two NN models. It is important to note that the dataset is split in a sequential manner. Specifically, the first 70% of the data is used for training, and the last 10% is used for testing. This is different from a random split into the 7:2:1 ratio. The reason for this sequential split is to ensure that the data in the testing dataset has not been seen by the models during the training process. This way, when the models are evaluated on the testing dataset (i.e., when making predictions on the testing set), it is truly testing the "extrapolation" performance of the models. In contrast, if the data was randomly split, there is a risk that the testing dataset might contain data that is similar to the training data, and in that case, the evaluation on the testing dataset would actually be testing the "interpolation" performance of the models, which is not the desired outcome for a comprehensive assessment of the models' generalization ability.

3. Results and discussion

3.1 *Model training*

In the present study, the open-source deep-learning library PyTorch is utilized to construct and train the NN models. PyTorch offers a wide range of tools and functions that are convenient for building and training neural networks. For the training process, the Adamax optimizer provided by PyTorch is employed. The Adamax optimizer is chosen because it has shown good performance in handling various types of neural network architectures and datasets. During each epoch of the training process, the snapshots in the training dataset are randomly shuffled. This random shuffling helps to prevent the neural network from over-fitting to the order of the data. The activation function used in the U-Net network is the rectified linear unit (ReLU) function, which is defined by:

$$f(x) = \max(0, x) \tag{10}$$

The loss function used for training is the root mean square error (E_{rms}) loss function, with its formula being:

$$E_{\text{rms}} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - y_i^{\text{pred}})^2} \tag{11}$$

In this paper, to make a fair comparison between different models and control the variables, each model is trained for the same number of epochs, specifically 1 000 epochs. After training for 1 000 epochs, the performance of these models is compared. Figure 5 shows the loss curve of one such training process. From the figure, it can be clearly observed that the loss of the neural network tends to converge after 800 epochs, indicating that the model is approaching an optimal state in terms of minimizing the error between the predicted and the actual values.

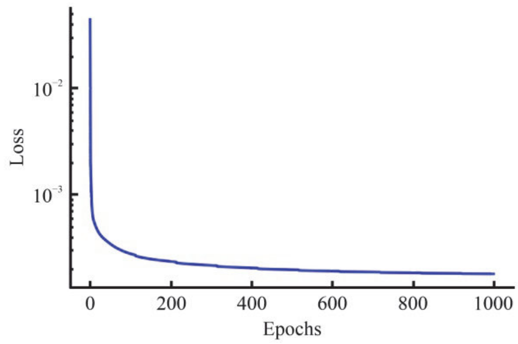


Fig. 5 Loss history of NNs in the training process

3.2 *NN predictions*

Once trained, the NN models are used for super-resolution reconstruction of the turbulent flow field. The reconstructed turbulent flow fields are shown in Fig. 6 (prediction on the training set) and Fig. 7 (prediction on the testing set). These figures provide a visual comparison of how the models perform on different subsets of the data, allowing us to assess their ability to generalize and reconstruct accurate flow fields.

By scrutinizing the above two figures, it can be clearly seen that both the U-Net and the pgU-Net models in this study perform significantly better than the bi-cubic interpolation method. However, it is difficult to directly distinguish the accuracy of the U-Net and pgU-Net models just by visually inspecting the images. Therefore, in this study, two commonly used methods for evaluating the accuracy of neural networks are employed. These methods are used to assess the accuracy of the three super-resolution models (bi-cubic interpolation, U-Net and pgU-Net) on the testing dataset, which is data that the neural networks have not seen during training. The two evaluation metrics are the root mean square error (E_{rms}) and the coefficient of determination (R^2). The definition of E_{rms} has been given in Eq. (10), and the formula for R^2 is given by:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - y_i^{\mathrm{pred}})^2}{\sum_{i=1}^{N}\left(y_i - \overline{y_i}\right)^2} \tag{12}$$

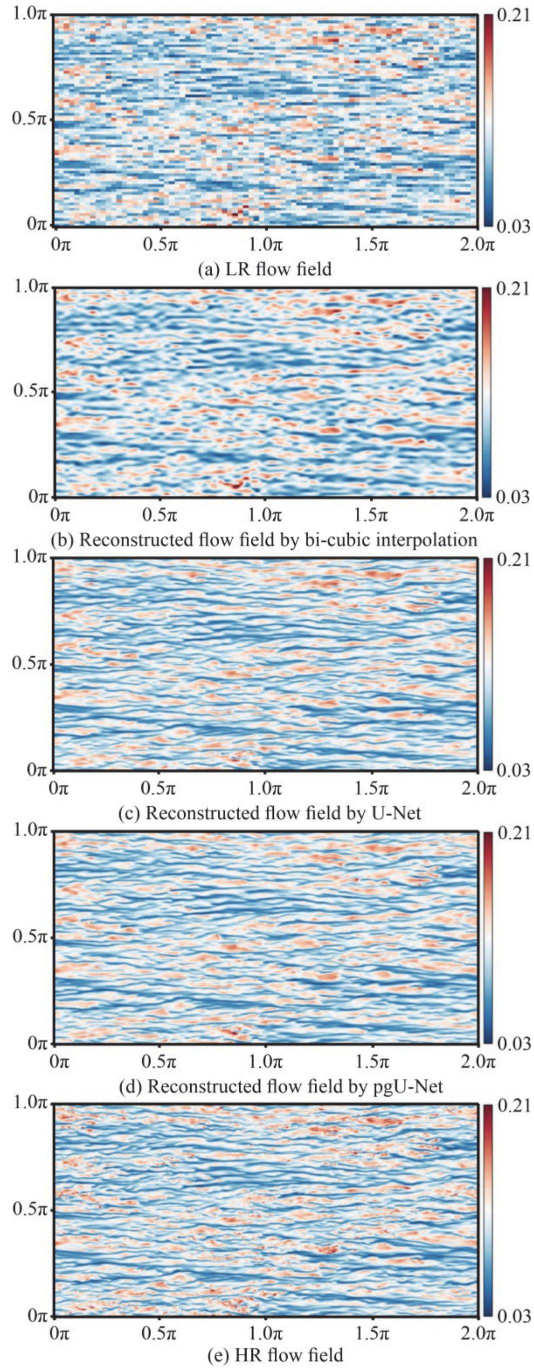


Fig. 6 training set predictions using the LR dataset with the DSR of 4

Tables 2, 3 present the prediction accuracies, in terms of E_{rms} and R^2 values, of the three super resolution models on the test set with a DSR of 4.

Table 2 E_{rms} **of SR flow field by using the LR dataset with the DSR of 4**

DSR	Bi-cubic	U-Net	pgU-Net
4×	0.024 780	0.011 753	0.010 006

Table 3 R^2 **score of SR flow field by using the LR dataset with the DSR of 8**

DSR	Bi-cubic	U-Net	pgU-Net
4×	0.062 003	0.893 580	0.920 649

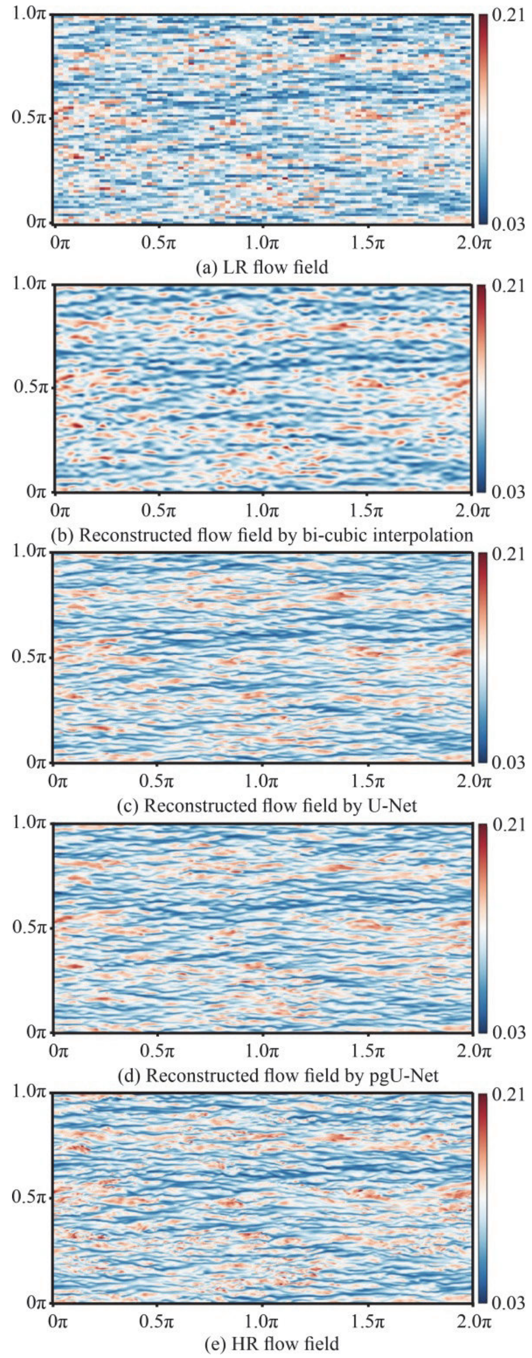


Fig. 7 testing set predictions using the LR dataset with the DSR of 4

From the data in these two tables, it is evident that both the U-Net and pgU-Net models far exceed the simple interpolation algorithm in terms of accuracy. Moreover, on the dataset with a DSR of 4, the R^2 value of the pgU-Net model shows a slight lead of 0.03 over that of the U-Net model, indicating that the pgU-Net has a marginally better performance in reconstructing the flow field.

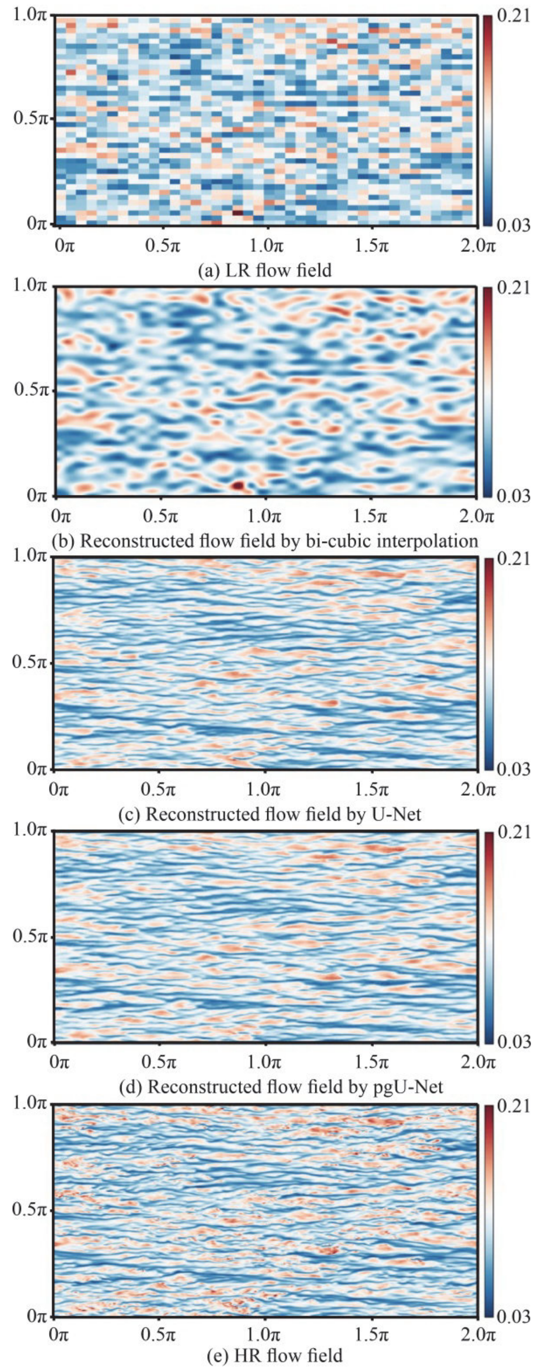


Fig. 8 Training set predictions using the LR dataset with the DSR of 8

Subsequently, to further examine the super resolution reconstruction performance of the neural network architectures, the DSR is increased. The U-Net and pgU-Net models are trained using data with a DSR of 8 and then used for super resolution reconstruction. The reconstructed turbulent flow fields are presented in Fig. 8 (prediction on the training set) and Fig. 9 (prediction on the testing set). This experiment with a higher DSR allows us to explore the models' capabilities under more challenging conditions, as a higher DSR means more information loss during down sampling, and thus a more difficult reconstruction task.

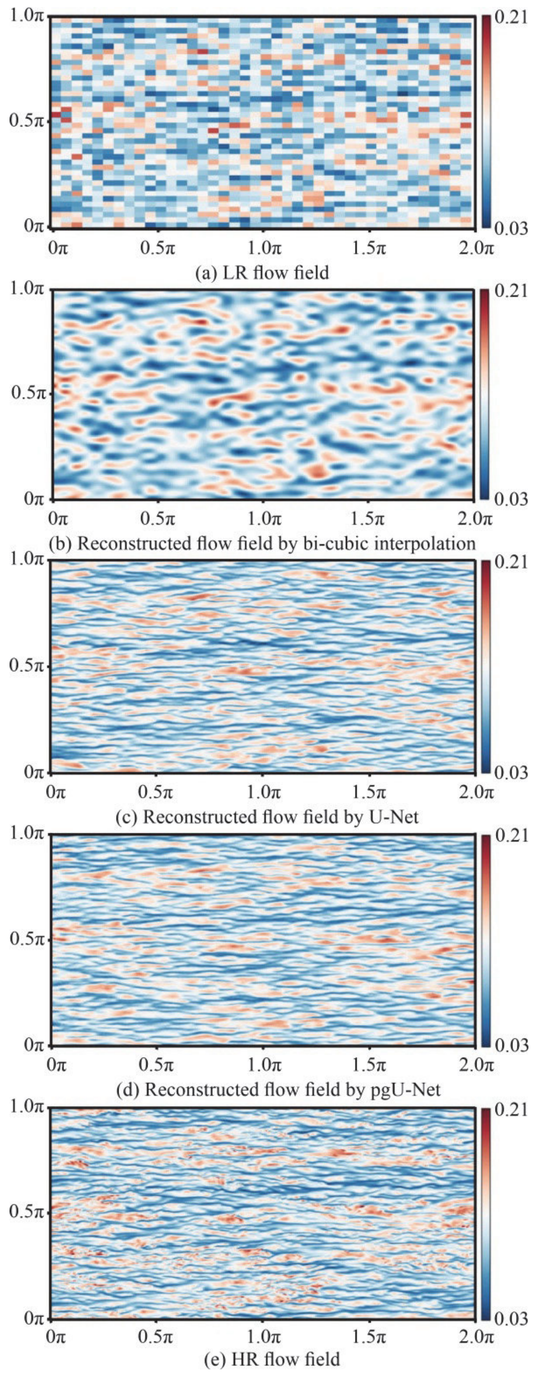


Fig. 9 Testing set predictions using the LR dataset with the DSR of 8

By presenting the cases with a DSR of 8, as shown in Figs. 8, 9, it can be clearly observed that the simple interpolation algorithm fails to reconstruct the flow field accurately. The reconstructed flow field using the bi-cubic interpolation is completely distorted. In contrast, the two NN-based super resolution models in this study, namely the U-Net and pgU-Net, are still able to successfully reconstruct the large-scale structures of the turbulent flow field. Although it is evident from visual inspection that there are some deficiencies in reconstructing the small-scale flow details, the fact that they can capture the large-scale features indicates their superiority over the simple interpolation method.

Further, through the calculation of E_{rms} and R^2 values, as presented in Tables 4, 5, it can be seen that the R^2 value of the simple bi-cubic interpolation algorithm is negative. This negative value indicates that the interpolation method is performing very poorly in reconstructing the flow field. The R^2 value of the U-Net neural network's prediction results is approximately 0.68, while the R^2 value of the pgU-Net's reconstruction results remains at 0.8 even on the test set of the dataset with a DSR of 8. This significant difference in R^2 values further validates that the pgU-Net model has a better performance in reconstructing the flow field, especially under more challenging down-sampling conditions.

Table 4 E_{rms} **of SR flow field by using the LR dataset with the DSR of 8**

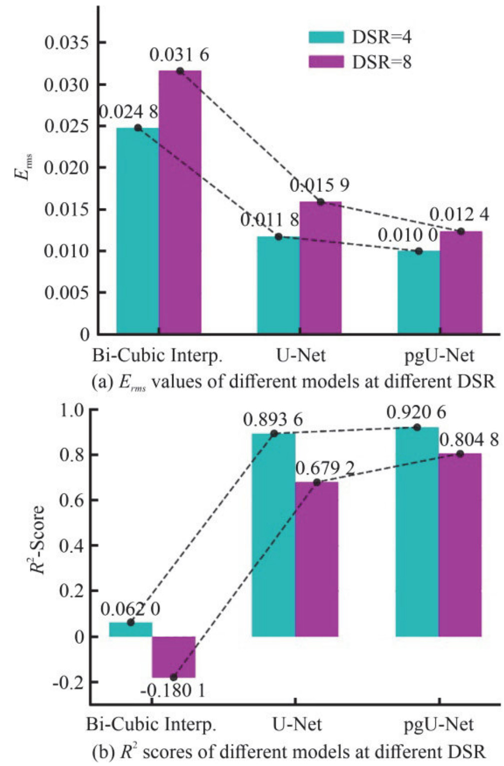
DSR	Bi-cubic	U-Net	pgU-Net
8×	0.031 597	0.015 892	0.012 368

Table 5 R^2 **score of SR flow field by using the LR dataset with the DSR of 8**

DSR	Bi-cubic	U-Net	pgU-Net
8×	–0.180 059	0.679 172	0.804 768

The data from Tables 1 to 4 are integrated and presented in Fig. 10. Through a detailed analysis of Fig. 10, the following observations can be made:

(1) Regarding the bi-cubic interpolation method, regardless of whether the DSR is 4 or 8, its performance is notably poor. Specifically, the corresponding root-mean-square error (E_{rms}) values are high, while the coefficient of determination (R^2) values are low. This indicates that the bi-cubic interpolation method alone is inadequate for performing super-resolution reconstruction of turbulent flow fields. In other words, it cannot effectively recover high-resolution flow field information from low-resolution data. The high E_{rms} implies a large deviation between the interpolated results and the actual high-resolution data, and the low R^2 further confirms the weak performance of the interpolation method in representing the real flow field characteristics.



(a) E_{rms} values of different models at different DSR

(b) R^2 scores of different models at different DSR

Fig. 10 Accuracy of different models at different DSR

(2) The U-Net neural network shows a significant improvement compared with the bi-cubic interpolation. Particularly when the DSR is 4, the $E_{\rm rms}$ of the U-Net neural network is 0.011 8, and the R^2 value approaches 0.84. This demonstrates that the U-Net neural network can better capture the features of the flow field and perform super-resolution reconstruction. However, when dealing with a turbulent flow field with a DSR of 8, the $E_{\rm rms}$ of the U-Net neural network increases from 0.011 8 to 0.015 9, and the corresponding R^2 value drops sharply from 0.84 to 0.68. This significant change suggests that when only the velocity field is used as input information, the U-Net neural network struggles to effectively perform turbulent super-resolution reconstruction on datasets with a DSR of 8. The increase in $E_{\rm rms}$ indicates a greater error in the reconstructed flow field, and the sharp decrease in R^2 reflects a reduced ability to approximate the real high-resolution flow field.

(3) The U-Net neural network with the addition of the pressure field channel, namely the pgU-Net in this study, exhibits a more remarkable improvement compared with the UNet neural network with only the velocity field channel (the U-Net in this study) on both $\mathrm{DSR}=4$, $\mathrm{DSR}=8$ datasets. On the $\mathrm{DSR}=4$ dataset, the R^2 value increases from 0.893 6 to 0.920 6. On the $\mathrm{DSR}=8$ dataset, the improvement is even more pronounced, with the R^2 value rising from 0.679 2 to 0.804 8. These increases in R^2 values clearly show that the pgU-Net can better reconstruct the turbulent flow field, reducing the error between the reconstructed and the actual high-resolution flow fields.

In conclusion, the above results comprehensively demonstrate the effectiveness and importance of incorporating the pressure field as an input feature in the task of turbulent super-resolution reconstruction. The pressure field provides additional physical information that helps the neural network better capture the global characteristics and correlations of the flow field, thus improving the accuracy and reliability of super-resolution reconstruction.

4. Conclusions

This work establishes a physics-guided deep learning framework for turbulent flow reconstruction that effectively integrates fluid mechanics principles with data-driven methodologies. By leveraging the elliptic nature of pressure fields in incompressible flows, the proposed pgU-Net architecture demonstrates superior SR performance compared with conventional approaches, particularly under aggressive down-sampling ratios. Key findings include:

(1) Pressure field integration improves reconstruction accuracy by 3.0% $(\mathrm{DSR}=4)$ to 12.6% $(\mathrm{DSR}=8)$ in R^2 metrics compared with pure data-driven U-Net implementations.

(2) The physics-guided approach maintains robust generalization capabilities, evidenced by consistent performance on unseen test data.

(3) Global pressure correlations effectively compensate for information loss in severely down sampled velocity fields.

These results highlight the critical importance of incorporating physical knowledge when applying machine learning to multiscale fluid dynamics problems. The demonstrated framework provides a computationally efficient alternative to conventional attention mechanisms while maintaining physical consistency.

Conflict of interest

All authors declare that there are no other competing interests.

References

[1] Jin Y., Chen J., Zhang J. et al. Large eddy simulations of zinc ions transfer to turbulent flows from hyporheic zone [J]. *Journal of Hydrodynamics*, 2024, 36(4): 650-661.

[2] Zhang B., Xu D., Ji C. et al. Investigation on the

width-to-depth ratio effect on turbulent flows in a sharp meandering channel with periodic boundaries using large eddy simulations [J]. *Journal of Hydrodynamics*, 2024, 36(4): 662-677.

[3] Fukami K., Fukagata K., Taira K. Machine-learning-based spatio-temporal super resolution reconstruction of turbulent flows [J]. *Journal of Fluid Mechanics*, 2020, 909: A9.

[4] Fukami K., Fukagata K., Taira K. Super-resolution reconstruction of turbulent flows with machine learning [J]. *Journal of Fluid Mechanics*, 2019, 870: 106-120.

[5] Fukami K., Taira K. Single-snapshot machine learning for super-resolution of turbulence [J]. *Journal of Fluid Mechanics*, 2024, 1001: A32.

[6] Liu B., Tang J., Huang H. et al. Deep learning methods for super-resolution reconstruction of turbulent flows [J]. *Physics of Fluids*, 2020, 32(2): 025105.

[7] Bi X., Liu A., Fan Y. et al. FlowSRNet: A multi-scale integration network for super-resolution reconstruction of fluid flows [J]. *Physics of Fluids*, 2022, 34(12): 127104.

[8] Xu Q., Zhuang Z., Pan Y. et al. Super-resolution reconstruction of turbulent flows with a transformer-based deep learning framework [J]. *Physics of Fluids*, 2023, 35(5): 055130.

[9] Zeng K., Zhang Y., Xu H. et al. Super-resolution reconstruction of turbulent flows with a hybrid framework of attention [J]. *Physics of Fluids*, 2024, 36(6): 065107.

[10] Sofos F., Drikakis D., Kokkinakis I. W. et al. Spatiotemporal super-resolution forecasting of high-speed turbulent flows [J]. *Physics of Fluids*, 2025, 37(1): 016124.

[11] Faroughi S. A., Pawar N. M., Fernandes C. et al. Physics-guided, physics-informed, and physics-encoded neural networks and operators in scientific computing: Fluid and solid mechanics [J]. *Journal of Computing and Information Science in Engineering*, 2024, 24(4): 040802.

[12] Ye M., Li M., Liu M. et al. Overview of data-driven models for wind turbine wake flows [J]. *Journal of Marine Science and Application*, 2025, 24(1): 1-20.

[13] Raissi M., Perdikaris P., Karniadakis G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations [J]. *Journal of Computational Physics*, 2019, 378: 686-707.

[14] Cai S., Mao Z., Wang Z. et al. Physics-informed neural networks (PINNs) for fluid mechanics: A review [J]. *Acta Mechanica Sinica*, 2021, 37(12): 1727-1738.

[15] Cuomo S., Di Cola V. S., Giampaolo F. et al. Scientific machine learning through physics–informed neural networks: Where we are and what's next [J]. *Journal of Scientific Computing*, 2022, 92(3): 88.

[16] Ye M., Zhong Y., Chen H. et al. Investigating the impact of numerical uncertainties in CFD-generated training data on neural network models for wind-turbine wake predictions [J]. *Ocean Engineering*, 2025, 343: 123200.

[17] Yousif M. Z., Yu L., Lim H. Physics-guided deep learning for generating turbulent inflow conditions [J]. *Journal of Fluid Mechanics*, 2022, 936: A21.

[18] Li B., Ge M., Li X. et al. A physics-guided machine learning framework for real-time dynamic wake prediction of wind turbines [J]. *Physics of Fluids*, 2024, 36(3): 035143.

[19] Huang Q., Zhu W., Ma F. et al. Sparse learning model with embedded RIP conditions for turbulence super-resolution reconstruction [J]. *Computer Methods in Applied Mechanics and Engineering*, 2024, 425: 116965.

[20] Nicoud F., Ducros F. Subgrid-scale stress modelling based on the square of the velocity gradient tensor [J]. *Flow, Turbulence and Combustion*, 1999, 62(3): 183-200.

[21] Fan G., Zhao W., Wan D. Near-wall turbulent fluctuations and coherent structures in wall-modeled large-eddy simulation [J]. *Physics of Fluids*, 2025, 37(9): 095108;

[22] Fan G., Chen H., Zhao W. et al. Scaling laws and space–time characteristics of wall pressure fluctuations in an axisymmetric boundary layer with varying pressure gradient [J]. *Journal of Fluid Mechanics*, 2025, 1016: A28.