Deep learning with fourier features for regressive flow field reconstruction from sparse sensor measurements

Phong Nguyen, Joseph Choi, Quang-Trung Luu

scientific reports

Check for updates

OPEN

Deep learning with fourier features for regressive flow field reconstruction from sparse sensor measurements

Phong C. H. Nguyen[1]✉, Joseph B. Choi[2] & Quang-Trung Luu[3]

Many applications in computational and experimental fluid mechanics require effective methods for reconstructing the flow fields from limited sensor data. However, this task remains a significant challenge because the measurement operator that provides the punctual sensor measurement for a given state of the flow field is often ill-conditioned and non-invertible. This issue impedes the feasibility of identifying the forward map, which is, theoretically, the inverse of the measurement operator, for field reconstruction purposes. While data-driven methods are available, their generalizability across different flow conditions (e.g., different Reynold numbers) remains questioned. Moreover, they frequently face the problem of spectral bias, which leads to smooth and blurry reconstructed fields, thereby decreasing the accuracy of reconstruction. We introduce FLRNet, a deep learning method for flow field reconstruction from sparse sensor measurements. FLRNet employs a variational autoencoder with Fourier feature layers and incorporates an extra perceptual loss term during training to learn a rich, low-dimensional latent representation of the flow field. The learned latent representation is then correlated to the sensor measurement using an attention-based network. We validated the reconstruction capability and the generalizability of FLRNet under various fluid flow conditions and sensor configurations, including different sensor counts and sensor layouts. Numerical experiments show that in all tested scenarios, FLRNet consistently outperformed other baselines, delivering the most accurate reconstructed flow field and being the most robust to noise.

The ability to rapidly reconstruct coherent flow fields from sparse sensor observations is essential in many areas of physical sciences and engineering, such as computational and experimental fluid dynamics, aerospace engineering, and climate science[1–5]. Mathematically, consider a complete flow field $\mathrm{u}(t)$ that returns values in $\mathbb{R}^m$ and a measurement operator $\mathscr{H} : \mathbb{R}^m \to \mathbb{R}^p$ that provides its punctual measurement $\mathrm{y}(t) \in \mathbb{R}^p$. The goal of the flow field reconstruction task is to find the forward map $\mathscr{G} : \mathbb{R}^p \to \mathbb{R}^m$, that is, in theory, the inverse of $\mathscr{H}$. However, finding the map $\mathscr{G}$ as the inverse of $\mathscr{H}$ is by no means straightforward. This is because the complete flow field u often has a higher dimension than its punctual measurement y, $m \gg p$; therefore, the measurement operator $\mathscr{H}$ is likely ill-conditioned and, thus, often invertible[6]. As a result, directly deriving $\mathscr{G}$ by inverting $\mathscr{H}$ is nearly impossible.

To tackle these problems, several approaches have been proposed to reconstruct the flow field from sparse sensor measurements. *Direct reconstruction*[7] refers to an approach that directly reconstructs the flow field through an optimization process. In this model-free method, the reconstructed fields are represented as a linear combination of reference modes, and optimization is used to derive their associated coefficients. *Data assimilation*[8] is another approach that involves the use of a dynamic model to predict the flow field from its previous states and use sensor measurements to improve forecast quality. Finally, *regressive reconstruction*, the focus of this paper, is an approach that uses machine learning (ML) methods to directly learn the forward map $\mathscr{G}$ from the data. Here, ML models are used to approximate the forward map, and the model parameters are derived by solving a training optimization problem in which the discrepancy between the machine-reconstructed field and the corresponding ground truth is minimized. Regressive reconstruction provides numerous computational efficiency benefits over direct reconstruction and data assimilation, as it allows for a single training and multiple reuses of the regression model with considerably faster computations.

[1]Faculty of Mechanical Engineering and Mechatronics, Phenikaa School of Engineering, Phenikaa University, Hanoi 100000, Vietnam. [2]School of Data Science, University of Virginia, Charlottesville, VA 22903, USA. [3]Université Paris-Saclay, CentraleSupélec, CNRS, L2S, 91190 Gif-sur-Yvette, France. ✉email: phong.nguyenconghong@phenikaa-uni.edu.vn

In line with regressive reconstruction, several notable works have successfully learned the forward map $\mathscr{G}$ to reconstruct the flow field from sensor measurements. Erichson *et al.*[9] used a shallow network to approximate the forward map $\mathscr{G}$ and trained it using data. The authors demonstrated a higher reconstruction accuracy for this method compared to traditional direct reconstruction methods based on POD in different validation cases. Li et al.[10] investigated different deep learning (DL) techniques for reconstructing chaotic flow fields with high Reynolds numbers. The models investigated include multilayer perceptron (MLP), convolutional neural networks (CNN), and generative adversarial networks (GAN). The results show that, while GAN is better at reconstructing small-scale vortex structures, MLP performs better with time-averaged flow fields and provides significant computational efficiency. Recent investigations, such as those of Wu et al.[3], Zhao et al.[11], and Xie et al.[12], highlight advances in regressive reconstruction of fluid flow fields using methods such as reduced order models, Fourier operators, and physics-informed learning. These studies have addressed, to some extent, critical challenges in reconstructing high-dimensional flow fields from sparse sensor data, particularly issues related to inadequate generalization due to limited data availability and convergence difficulties from vanishing or exploding gradients. Although various solutions have been suggested, for example, L1/L2 regularization and batch normalization[9], these challenges persist.

The root cause of the difficulty in reconstructing flow fields from sparse sensor measurements stems from the considerable dimensional disparity between the observational space of the sensor measurements and the state space of the original flow fields. This discrepancy renders the forward map $\mathscr{G}$ difficult to directly approximate due to ill-posed reconstruction problems. To address this challenge, the use of dimensionality reduction has been proposed to reduce the difference in input and output dimensions to promote a better understanding of the forward map $\mathscr{G}$. In this approach, the first objective is to find an efficient low-dimensional representation z of the flow field u. Consequently, ML is used to learn the map $\mathscr{G}_z$ between the sensor measurements y and the learned low-dimensional representation z. Learning to map sensor observations to a low-dimensional representation (i.e., learning $\mathscr{G}_z$) can be more effective than learning the direct map $\mathscr{G}$ for two reasons. First, the sample complexity of a regression task scales with its output dimension, so compressing the target flow field from m to $k \ll m$ reduces the amount of data required for a given error bound, thus making training easier to converge. Second, the reduced problem is better conditioned, as most ill-posed high-frequency modes are absorbed by the decoder that later lifts z back to the full field; thus, it is easier to solve. In line with this thought, Dubois et al.[6] have provided empirical supports for this hypothesis by investigating the use of several dimensionality reduction techniques, including both linear and nonlinear ones, to learn the low-dimensional representation of the flow field. The authors showed that by reducing the dimension of the output using latent representations and selecting the appropriate regression model, an efficient and accurate reconstruction model can be obtained.

Despite the potential of the dimensionality reduction approach, most methods validate only with a single flow condition (e.g., a single Reynold number)[6,9,11], leaving their generalizability across different flow conditions unknown (see Table 1 for more details). In addition, *spectral bias* is another critical problem in physics-based representation learning that needs to be overcome. This is the phenomenon in which the ML models are biased toward global flow features with low frequency while commonly missing the local ones with high frequency. This issue causes the reconstructed field to smooth out, thus reducing its accuracy. In fact, this is a common problem in ML for physical sciences and engineering but has not been adequately addressed[13,14].

In this work, we introduce FLRNet, a DL method for flow field reconstruction from sensor measurements that can accommodate the above issues. FLRNet consists of two DL models: (i) a Fourier feature-based variational autoencoder (VAE), trained with added perceptual loss to learn the low-dimensional representations of flow fields; and (ii) a sensor encoding network to correlate the learned representation with the corresponding sensor measurements. During inference, the sensor encoding network will derive the latent representation of the flow field given its sensor measurements, and the trained decoder will reconstruct the flow field from the derived latent representation.

Main contributions: The main contribution of FLRNet is twofold. *First*, we introduce Fourier feature layers that help mitigate the problem of low frequency bias, improving the accuracy of the model reconstruction compared to other baselines, as will be shown later. *Second*, we trained and tested FLRNet using a single canonical geometry in a wide range of flow regimes with various flow conditions. As mentioned above, this aspect is crucial for ensuring the practicality of the method, yet it has not received sufficient attention in prior works.

The remainder of the paper is structured as follows. We first provide some background on the related work in the field. Consequently, we introduce the proposed method and its validation with a legacy benchmark problem, namely, reconstructing the flow field around a circular obstacle. Here, we also assess the effects of several elements on the accuracy of the reconstructed flow field, including the number of sensors used and the

References	Flow type	Reynolds number(s)
Dubois et al.[6]	Circular cylinder flow	200, 500, 20,000 (multiple)
Wu et al.[3]	Circular cylinder flow	100 (single)
Zhao et al.[11]	Circular cylinder flow	100 (single)
Erichson et al.[9]	Circular cylinder flow	100 (single)
Li et al.[10]	Turbulent wake of square	22,000 (single)
This work (FLRNet)	Circular cylinder flow	10–10,000 (multiple)

Table 1. Summary of flow types and Reynolds numbers addressed by previous works and the current study.

layout of the sensor placement. Finally, we discuss the advantages and disadvantages of the proposed method compared to other baseline methods and suggest several directions for improvement.

Related work

Flow field reconstruction from sensor measurements

There are three common approaches to reconstruct the flow field from sparse sensor measurements[6]. *Direct reconstruction*[7] refers to an approach that directly reconstructs the flow field through an optimization process. This is a model-free method in which the reconstructed fields are represented as a linear combination of reference modes, $\mathrm{u} \approx \hat{\mathrm{u}} = \sum_{j=1}^{k} \phi_j \nu_j$, where ϕ_j and ν_j $(j \in [1, k])$ are, respectively, the reference modes and the associated coefficients. Here, modes ϕ_j can be approximated using modal decomposition such as POD[15] or sparse representation[16]. The goal of the direct reconstruction approach is to find the coefficient vector ν_j that minimizes the discrepancy between the reconstructed and the ground truth flow field: $\nu = \mathrm{argmin}_{\nu} \|\mathscr{H}(\mathrm{u}) - \mathscr{H}(\sum_{j=1}^{k} \phi_j \nu_j)\|_2^2$.

The second approach, namely *data assimilation*, is another method that is often used for field reconstruction from sensor measurements. This approach involves the use of a dynamical model to estimate the field. Here, sensor measurements are used to improve forecast quality. The dynamical model can be a reduced-order model[17] or a DL-based model[14,18,19]. The drawback of such methods is that the accuracy of the reconstructed fields is heavily dependent on the quality of the dynamical model, which is typically hard to achieve. As the field evolves temporally, the accuracy of the dynamical model will decrease as a result of error accumulation.

Finally, *regressive reconstruction*[9,20] is the third approach that uses ML methods to directly learn the forward map $\mathscr{G}$ from the data instead of solving an optimization problem as in direct reconstruction. Here, the ML model is used to approximate the map $\mathscr{G}$, such that $\mathscr{G}(\mathrm{y}|\theta)$ where θ are model parameters. The training goal here is to find the optimal set of θ that minimizes the discrepancy between the machine-reconstructed field and the corresponding ground truth: $\theta = \mathrm{argmin}_{\theta} \|\mathrm{u} - \mathscr{G}(\mathrm{y}|\theta)\|_2^2$. Regressive reconstruction offers many advantages in computational efficiency compared to direct reconstruction, as the regression model can be trained once and reused multiple times.

It is important to recognize the distinction between direct reconstruction and regressive reconstruction, despite their shared objective of addressing an optimization problem aimed at reconstructing the original flow field. In the case of direct reconstruction, commonly referred to as "online" methods, no predictive model undergoes prior training. Rather, the algorithm autonomously resolves an inverse problem in real time for each novel set of sensor measurements by selecting a set of coefficients $\boldsymbol{\nu}$ that, when integrated with a fixed basis (such as POD modes $\{\boldsymbol{\phi}_j\}$), best aligns with the observed data through the sensor operator H. Conversely, regressive reconstruction, also known as "off-line" methods, constructs a parametric map $G_{\boldsymbol{\theta}} : \mathrm{y} \mapsto \hat{\mathrm{u}}$ from sensor measurements to full-state predictions through direct supervised learning on paired training data. This methodology effectively integrates both the representation (basis) and the coefficient inference within the learned weights $\boldsymbol{\theta}$, facilitating rapid inference through a singular forward computation. Although variables like ϕ and θ can appear functionally similar, fundamental differentiation is found in the manner and timing of the inversion process—whether through instance-specific optimization or learned regression—as well as the nature of the supervision involved. In this work, we focus on regressive reconstruction as a result of its computational advantages compared to other methods.

Deep learning for regressive reconstruction

Recently, DL has found widespread application in field reconstruction from sparse sensor measurements. Erichson et al.[9] were among the first to propose the use of DL for regressive reconstruction. The authors used a model known as the shallow network to directly learn the forward map from the observational space to the state space. The shallow network demonstrated an improvement in reconstruction accuracy compared to traditional direct reconstruction methods, including the standard improved POD-based method. Peng et al.[21] proposed a hybrid DL framework that combines traditional mode decomposition with residual learning using MLP networks to enhance the accuracy of field reconstruction from sparse sensor measurements of unsteady periodic flow fields.

Although the early successes of DL in flow field reconstruction are significant, the notable dimensional disparity between observation and state spaces remains a major hurdle in this field. Addressing this dimensional gap requires advanced neural network architectures that can effectively capture the complex nonlinear relationships between sparse sensor data and comprehensive flow fields. However, researchers often lack adequate training datasets to learn these intricate mappings. Due to the limited availability of training data compared to the high model complexity necessary, DL models are prone to overfitting to the specific flow patterns present in the training data, as opposed to learning broadly applicable fluid dynamic principles. Therefore, this data scarcity results in only suboptimal reconstruction performance for flow configurations that are absent from the training data. Furthermore, reconstructing flow from sparse observations is inherently an ill-posed inverse problem, since different flow states may produce identical sensor output. This non-uniqueness creates ambiguous input-output relationships within the training samples, making it particularly difficult for neural networks to learn consistent reconstruction patterns. Combined with data scarcity and the high-dimensional nature of the reconstruction task, these factors increase the risk of overfitting DL models and pose challenges in achieving convergence, thereby requiring careful regularization strategies.

For this reason, Dubois et al.[6] proposed the use of dimension reduction techniques to resolve the large difference in the two spaces. The authors investigated the use of different dimensionality reduction techniques, including linear and nonlinear ones. The results showed that the deep variational autoencoder yields the highest

reconstruction accuracy and is the most robust to noise included in the sensor measurements. In addition to the use of MLP networks, researchers have also pursued the use of CNNs for field reconstruction from sensor measurements. Building upon success in computer vision, many efforts have been made to apply CNNs for field reconstruction. For example, Liu et al.[22] combined residual CNNs with MLPs to reconstruct the flow field from various measurable information. In another work, Peng et al.[23] proposed the use of U-net for the reconstruction of temperature field in heat sink designs. The authors represented the sensor value and position with a two-dimensional map, transforming the field reconstruction problem into an image-to-image translation problem, which was then solved using U-net. The use of neural operators, with the discretization-invariant characteristic, is also another line of potential research that has just been explored. As demonstrated by Zhao et al.[11], neural operators can provide zero-shot super-resolution while achieving similar or even better reconstruction accuracy compared to traditional CNN-based methods. This implies that neural operators can now be trained on low-resolution data while still making inferences on high-resolution data. This capability of neural operators helps reduce the size of required training data, which is extremely valuable given the complexity of collecting scientific data.

Despite certain successes of earlier research, as mentioned above, most of these methods were only validated for a limited range of flow conditions, and retraining is required if a new flow condition is required. Secondly, as with the majority field prediction problems in ML for science, spectral bias is still a big concern for most methods, as it reduces the accuracy of the reconstructed field both visually and quantitatively. The literature frequently reports that predicted fields tend to smooth out, particularly at the boundaries between phases[14,19]. In this work, our aim is to overcome these aforementioned issues.

Methods

Problem formulation

Figure 1a illustrates our problem formulation. In a finite-dimensional space, given the snapshot of the flow field $\mathrm{u} \in \mathbb{R}^m$ at a given time t and the map $\mathscr{H} : \mathbb{R}^m \to \mathbb{R}^p$ that provides the punctual measurement y of u, our goal is to find the forward map $\mathscr{G} : \mathbb{R}^p \to \mathbb{R}^m$ which is, in theory, the inverse of $\mathscr{H}$, to reconstruct u from y:

$$\hat{\mathrm{u}} = \mathscr{G}(\mathrm{y}) \approx \mathscr{H}^{-1}(\mathrm{y}) \tag{1}$$

where $\hat{\mathrm{u}}$ is the reconstructed flow field.

Because the dimension of u is much higher than that of y, trying to directly learn $\mathscr{G}$ from the data can be complicated and difficult, as mentioned above. Therefore, we first aim to find the low-dimensional latent representation z of u to reduce the dimensional difference of the map between the observational and the state spaces. This representation learning task is accomplished by employing a variational autoencoder (VAE).

VAE is a bottleneck-like neural network architecture consisting of two networks, namely the encoder $\mathscr{E}$ and the decoder $\mathscr{D}$. In VAE, the encoder $\mathscr{E}$ learns to encode the flow field u into a latent representation z:

$$\mathrm{z} = \mathscr{E}(\mathrm{u}). \tag{2}$$

Meanwhile, the decoder $\mathscr{D}$ learns to accurately reconstruct the corresponding flow field u from a given latent representation z:

$$\hat{\mathrm{u}} = \mathscr{D}(\mathrm{z}) = \mathscr{D} \circ \mathscr{E}(\mathrm{u}). \tag{3}$$

The probabilistic approach of VAE relies on two well-established components, which will be discussed later in the '*Architecture design*' section, namely (1) the Kullback–Leibler divergence loss term, which regularizes the latent distribution toward a unit Gaussian, ensuring a smooth and structured latent space, and (2) the reparameterization trick, which enables stable training by injecting stochasticity into latent codes[24]. These mechanisms ensure that the encoder and decoder learn representations where similar inputs map to nearby latent codes, and the decoder reconstructs accurately across a distribution of latent samples.

This probabilistic learning strategy for VAE presents a few advantages. *First*, this learning strategy forces both the encoder $\mathscr{E}$ and the decoder $\mathscr{D}$ to develop the capability to learn strong and robust features that effectively represent the ground truth full-state data $\hat{\mathrm{u}}$, enhancing the accuracy of the reconstruction process. *Second*, the latent space of VAEs is continuous and their latent variables are well distributed over the latent space. This smooth latent structure is particularly beneficial in the context of flow field reconstruction, where a regression model predicts the latent code from sparse sensor measurements, as it enhances robustness to perturbations and supports effective reconstruction of complex flow fields.

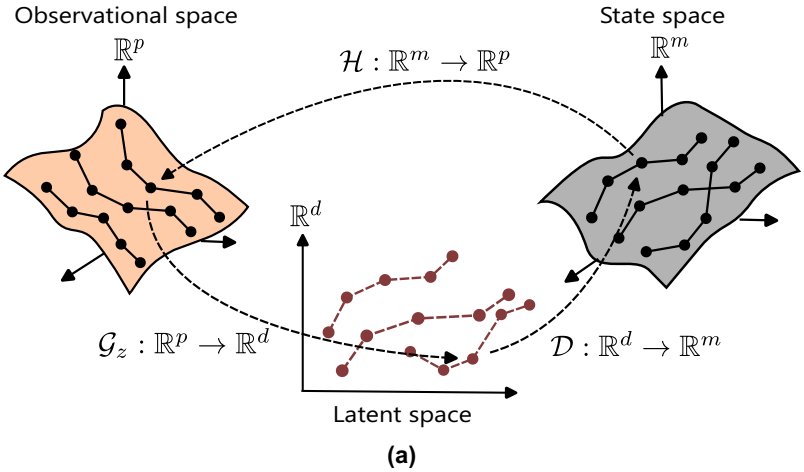
After successfully learning the representation of the target flow field, we attempt to learn the map that correlates sensor measurements y to the learned low-dimensional latent representation z of u:

$$\mathrm{z} = \mathscr{G}_z(\mathrm{y}). \tag{4}$$

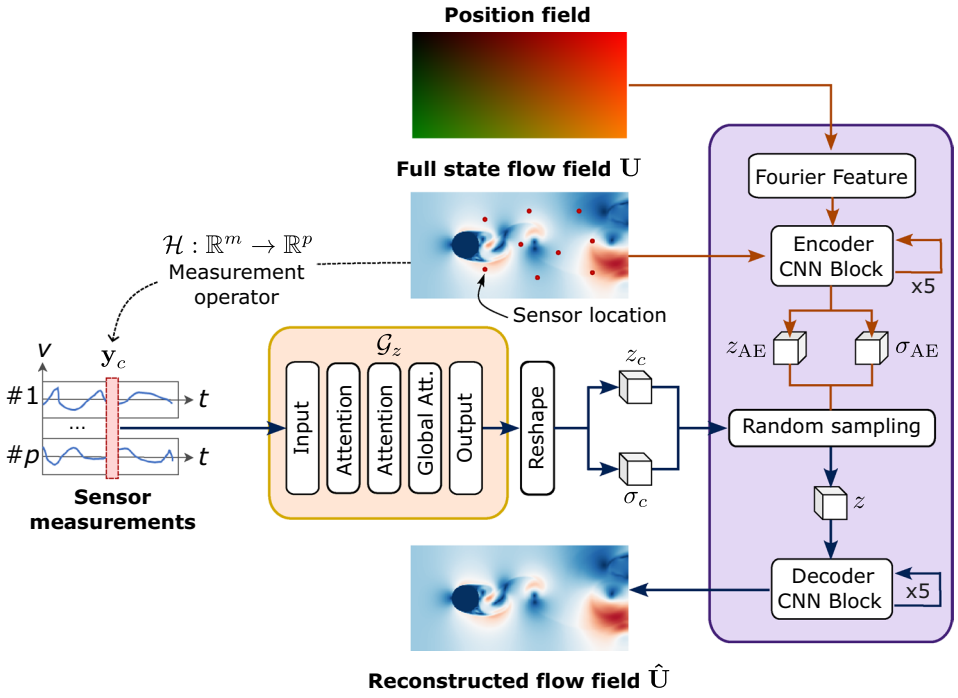
Finally, the reconstructed $\hat{\mathrm{u}}$ will be derived from the computed z through the learned decoding process:

$$\hat{\mathrm{u}} = \mathscr{D} \circ \mathscr{G}_z(\mathrm{y}) \tag{5}$$

where $\mathscr{G}_z$ is the map from y to z and $\mathscr{D}$ is the map from z to u, which is the decoding process. Following the regressive reconstruction method and inspired by the universal approximation theorem, we model $\mathscr{E}$, $\mathscr{D}$, and $\mathscr{G}_z$ using neural networks and try to learn them from data.



(a)



(b)

Fig. 1. Problem formulation and architectural design of FLRNet. (**a**) *Problem formulation*: Instead of directly learning the mapping from the observational space to the state space, FLRNet first maps the sparse sensor measurements into a low-dimensional latent representation and then reconstructs the original flow field using a pretrained decoder. (**b**) *Architecture design:* FLRNet consists of two primary components. The VAE (purple block) learns a compact yet expressive latent representation of the full flow field, while the Sensor Encoding Network (orange block) maps sensor measurements to this corresponding latent representation.

Architecture design

Figure 1b presents the overarching design of FLRNet, which is the realization of the mathematical formulation described above. The proposed DL architecture is composed of two primary components. The first component (purple block) utilizes a VAE to acquire a low-dimensional representation of the original flow field. The second component (orange block) applies an attention-based network to map the sensor measurements to the latent representation of the associated flow field.

For the VAE, we used a fully convolutional architecture based on Fourier features, trained with an additional perceptual loss term. While the use of Fourier features allows the network to capture multiscale spatial patterns more easily, the application of perceptual loss ensures that the reconstructed flow fields preserve the crucial visual features of flow fields. These architecture design choices allow the VAE to learn rich latent representations, thus improving the reconstruction accuracy, as will be demonstrated later in the results section.

During inference, the encoder $\mathscr{E}$ is discarded and only the decoder $\mathscr{D}$ and the map $\mathscr{G}_z$ are retained for reconstruction of the flow field from the measurements from the sparse sensor.

Variational autoencoder: The first important component of our framework is to learn the rich, low-dimensional latent representation of the flow field via a VAE.

VAE[24] is a special type of autoencoder which contains two networks: encoder and decoder. In VAE, the encoder $\mathscr{E}(\phi)$ maps the original full-state data $\mathbf{u}$ to a latent distribution $\mathrm{z} \sim q(\mathbf{z})$, typically a Gaussian distribution. Meanwhile, the decoder $\mathscr{D}$ reconstructs the original flow field $\hat{\mathrm{u}}$ from the latent distribution generated by the encoding process $\mathrm{z} \sim q_\phi(\mathbf{z})$.

Figure 1b illustrates our VAE architecture. We used a fully convolutional architecture design with the input given in a uniform grid of size 128×256. We first use five consecutive convolutional blocks to gradually reduce spatial dimensions by half after each block and produce 64, 128, 128, 256, and 256 feature maps, respectively, resulting in a final spatial size of 4×8. Using the common reparameterization trick in VAE training, we used two 1×1 convolution to compress the final feature map into two 8-dimensional maps that represent latent variable means and variances. Our decoder uses a symmetric architecture to reconstruct the original flow field from sampled latent variables.

Fourier feature: The first notable improvement we made to the architecture design of FLRNet is the use of Fourier feature mapping to enhance the ability of FLRNet to capture high-frequency flow features. Fourier feature mapping[25] is a common technique in DL for positional encoding in many applications. Previous literature has shown that Fourier features can effectively address spectral bias[26], improving the learning capabilities of physics-informed DL models[13]. The effectiveness of Fourier features against the spectral bias issue is due to their ability to control the frequency drop, which is the primary factor that prevents the network from learning high-frequency features[26]. In our work, we extend the traditional Fourier feature for MLP networks to adapt it to our fully convolutional architecture, as described below.

The Fourier features can be extracted from a position field which denotes the horizontal and vertical coordinates of a grid point within the examined domain. First, the position fields $\mathrm{x} := [\mathrm{x}_1, \mathbf{x_2}]$ are inputted into the encoder, along with the original flow field. Here, $\mathbf{x_1}$ and x_2 are the horizontal and vertical coordinates of the corresponding pixel. Note that $\mathbf{x_1}$ and x_2 have the same dimension as the original flow field. Consequently, we apply a Gaussian Fourier mapping γ pixel-wise:

$$\gamma(\mathrm{x}) = [\cos(\mathrm{B}x_1), \sin(\mathrm{B}x_1), \cos(\mathrm{B}x_2), \sin(\mathrm{B}x_2)] \tag{6}$$

where $\mathrm{B} \in \mathbb{R}^{m_f \times 1}$ is randomly sampled from the normal distribution $\mathscr{N}(0, \sigma_f^2)$. Here, the number of mapping features m_f and the standard deviation σ_f are the hyperparameters that are preset prior to training. In our experiments, we set $m_f = 4$ and $\sigma_f = 5$.

The Fourier features are computed once from the spatial coordinates and concatenated with the output of each convolutional block. To maintain dimensional consistency as feature maps are downsampled through the network, the Fourier features are correspondingly reduced in spatial resolution using max-pooling operations.

Sensor encoding network: The second part of our framework involves learning the mapping $\mathscr{G}_z$ that correlates the sensor measurements y to their corresponding low-dimensional latent representation z. Figure 1billustrates the network architecture used for $\mathscr{G}_z$, which employs an attention-based design inspired by its strong modeling capability.

First, the sensor measurement vector $\mathrm{y}(t)$ is passed through a positional embedding layer, transforming the raw sensor readings into a unified hidden-dimensional representation. Subsequently, two transformer-inspired attention blocks are applied sequentially to process the data. Each attention block comprises a multihead self-attention layer followed by a two-layer feedforward network using GELU activations, residual connections, and layer normalization. Following these attention blocks, a global attention pooling mechanism using single-head attention dynamically weights sensor contributions based on the current input to derive the final features. Finally, the output features from the global attention are projected onto a 256-dimensional vector, which is then reshaped into two $4 \times 8 \times 8$ tensors, representing the latent variable's mean and variance.

Learning objectives

Two stage training strategy: FLRNet was trained using a two-stage training strategy. In the first stage, only the VAE was trained. The goal of this stage is to learn a rich latent representation of the original flow fields. In the second stage, the mapping network $\mathscr{G}_z$ was trained to learn to correlate sensor measurements with their corresponding latent representation of the flow field. Note that in the second stage, we freeze the trained parameters of both the encoder and decoder.

VAE pre-training: VAE training is achieved by solving an optimization problem for the encoder and decoder network parameters, ϕ and θ, respectively. Here, the loss function below is minimized:

$$\mathscr{L}_{\mathrm{VAE}}(\phi, \theta) = -\mathbb{E}_{\mathrm{z} \sim q_\phi(\mathrm{z}|\mathbf{u})} \left[\log p_\theta(\hat{\mathrm{u}}|\mathrm{z}) + d_{\mathrm{KL}}(q_\phi(\mathrm{z}|\mathbf{u}) \| p_\theta(\mathrm{z})) \right]. \tag{7}$$

In (7), $\mathscr{L}_{\mathrm{rec}} = \mathbb{E}_{\mathrm{z} \sim q_\phi(\mathrm{z}|\mathbf{u})}(\log p_\theta(\hat{\mathrm{u}}|\mathrm{z})$ is the reconstruction loss, which aims to guarantee the accuracy of the reconstructed field $\hat{\mathrm{u}}$ compared to the field of ground truth $\mathbf{u}$. In VAE, the reconstruction loss is typically modeled as a Gaussian distribution ($\mathscr{N}(\hat{\mathrm{u}}; \mu_\theta(\mathrm{z}), \sigma^2 I)$), with fixed variance ($\sigma^2$) for computational simplicity[24]. Therefore, minimizing the reconstruction loss is equivalent to minimizing the negative log-likelihood, which, under the Gaussian assumption, reduces into:

$$\mathscr{L}_{\mathrm{rec}} \propto \frac{1}{2\sigma^2} |\mathrm{u} - \hat{\mathrm{u}}|^2 + \mathrm{const.} \tag{8}$$

This expression is proportional to the mean squared error (MSE) between the ground truth field (u) and the reconstructed field (û); therefore, MSE was used as a practical implementation of reconstruction loss.

The other part of $\mathscr{L}_{\mathrm{VAE}}$ is the d_{KL}. This is the Kullback–Leibler divergence that measures the distance between the latent distribution produced by the encoder $\mathscr{E}$, $q_\phi(\mathrm{z}|\mathbf{u})$, and the target latent distribution $p_\theta(\mathrm{z})$. In the case of VAE, a normal distribution $\mathscr{N}(0,1)$ was used as the target latent distribution, following the common convention in VAE training[24].

Sensor encoding network training: Once the VAE had been trained, we continued to train the sensor mapping network. This is also an optimization process, in which the most optimal network parameters γ would be found to minimize the following loss function:

$$\mathscr{L}_{\mathrm{mapping}}(\gamma) = -\mathbb{E}_{\mathrm{z}\sim\mathscr{G}_z(\mathrm{z}|\mathrm{y})}\left[\log p_\theta(\hat{\mathrm{u}}|\mathrm{z}) + d_{\mathrm{KL}}\left(\mathscr{G}_z(\mathrm{z}|\mathrm{y})\|q_\phi(\mathrm{z}|\mathbf{u})\right)\right]. \tag{9}$$

The training loss function presented in Eq. (9) comprises two distinct terms, similar to those employed in VAE pre-training. The first term $\mathbb{E}_{\mathrm{z}\sim\mathscr{G}_z(\mathrm{zy})}\left(\log p_\theta(\hat{\mathrm{u}}\mathrm{z})\right)$ constitutes the reconstruction loss component, which is implemented by means of the mean squared error (MSE). The second term $d_{\mathrm{KL}}\left(\mathscr{G}_z(\mathrm{zy})\|q_\phi(\mathrm{zu}\right)$ is designed to minimize the divergence between two probability distributions: the latent representation predicted by the sensor mapping network $\mathscr{G}_z(\mathrm{zy})$ and that derived by the VAE encoder $q_\phi(\mathrm{zu})$. This formulation of the loss function ensures that the sensor mapping network not only reconstructs flow fields with precision but also correctly maps sensor measurements to their respective latent representations.

Perceptual loss: In addition to VAE and sensor encoding training loss, we also introduce perceptual loss into the training to enhance the model capability to capture high-fidelity features of the flow field. Perceptual loss is a common loss function in the computer vision community to enable deep-generative models to produce high-fidelity synthetic images[27]. The goal of the perceptual loss function is to guide the model in learning to reconstruct the target snapshot image with a high level of fidelity and low discrepancy not only in the pixel space, but also in a feature space. The use of loss functions defined on the feature space will overcome the lack of 'perceptual similarity measurement of per-pixel space and allow the network to learn to extract stronger and more meaningful features. Furthermore, the incorporation of perceptual loss will reduce the appeal of low-frequency modes in neural network training, thereby directing convergence toward more precise solutions.

The perceptual discrepancy between the two flow fields can be computed as

$$\mathscr{L}_{\mathrm{perceptual}} = \|\Phi(\hat{\mathrm{u}}) - \Phi(\mathrm{u})\|_2^2 \tag{10}$$

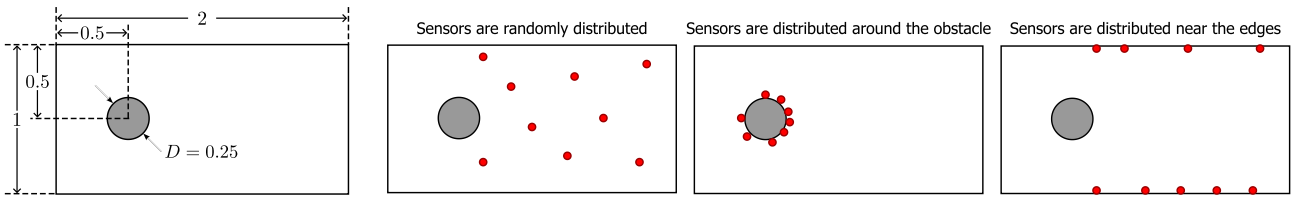
where u is the reconstructed flow field, û is its corresponding ground truth, and $\Phi(.)$ is the activation map in the final convolution layer of a pre-training deep neural network[27]. In this work, we use Inception-V3[28], which was pre-trained with the Imagenet dataset[29].

Training hyperparameter: In both stages, Adam Optimizer was employed with an initial learning rate of 1×10^{-5}. In addition, we also applied several techniques to increase the stability of the training. In particular, learning rate reduction is activated if validation loss does not improve for 15 consecutive epochs, with a reduction rate of 0.5 to a minimum of 1×10^{-7}. Early stopping was also applied if there was no improvement in the model after 50 epochs. All experiments used a batch size of 8 and were trained for a maximum of 200 epochs on an NVIDIA RTX 2080Ti GPU.

Data prepration

Flow data: We validate FLRNet using the canonical problem of flow around a cylindrical obstacle, which is a fundamental benchmark in computational fluid dynamics. In this work, we consider a broad range of flow complexity with Reynolds numbers ranging from 10 to 10, 000. This range covers several distinct flow characteristics: steady laminar flow ($Re < 47$), unsteady vortex shedding ($47 < Re < 300$) and transition to highly unstable and chaotic flow ($Re > 300$). Though the geometry of the examined domain is fixed, the variation of flow velocity creates multiple benchmark scenarios that will thoroughly validate FLRNet's ability to handle different flow physics from steady laminar to chaotic flow.

The flow simulations were conducted in a rectangular domain $2 \times 1\,\mathrm{m}$ with a circular obstacle of diameter $D = 0.25\,\mathrm{m}$, centered $0.5\,\mathrm{m}$ from the left and top boundaries (Fig. 2a). The inlet boundary condition was set to a uniform free-flow velocity of $u_{\mathrm{in}} = 1.0\,\mathrm{m/s}$. To vary the Reynolds number while keeping the inlet velocity and geometry fixed, the fluid density ρ was defined as $\rho = \frac{Re\,\mu}{u_{\mathrm{in}} D}$,, where $\mu = 1\,\mathrm{Pa.s}$ is the dynamic viscosity. This



(a) Domain's dimension. **(b)** Three different tested sensor layouts.

Fig. 2. Numerical experiment setting for the flow around cylindrical test problem. (**a**) Dimension of the examined domain. (**b**) Three different tested sensor layouts.

yields a kinematic viscosity of $\nu = \mu/\rho = \frac{u_{\text{in}} D}{Re}$. This parameterization is consistent with standard practices in fluid dynamics, allowing systematic variation of *Re* under fixed velocity and geometry. Each simulation was run for a total physical time of $2.0\,\text{s}$ and divided into 40 snapshots with a time step $\Delta t = 0.05\,\text{s}$. The chosen duration ensured capture of characteristic flow behaviors, including start-up transients and the transition to steady or periodic states, across all Reynolds numbers tested. In this problem, we are interested in reconstructing the velocity field from the velocity measurement data.

There are 39 simulation cases in the data set, 31 of which were used for training and 8 for testing. The Reynolds numbers used for training are $Re_{\text{train}} = [15, 20, 40, 80, 100, 140, 150, 200, 250, 350, 400, 450, 500, 550, 600, 650, 750,$ 800, 850, 900, 1000, 2000, 4000, 5000, 6000, 7000, 9000, 10000]. Meanwhile, the Reynolds numbers used for the tests are $Re_{\text{test}} = [30, 60, 120, 300, 700, 950, 3000, 8000]$.

Sensor measurement data: In the examined domain, we randomly place sensors to observe the flow velocity and aim to reconstruct the velocity field from these sensor measurements. The sensor placement was completed using the Latin hypercube method. In this experiment, we validated FLRNet with cases where 8, 16, and 32 sensors were used. We also tested three different sensor layouts: sensors distributed randomly throughout the domain, sensors distributed around the cylindrical obstacle, and sensors distributed near the boundary of the wall of the domain (Fig. 2b).

Validation metrics

Mean absolute error (MAE): To quantify reconstruction accuracy, we report the *mean absolute error* (MAE):

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} \left| \hat{u}_i - u_i \right| \tag{11}$$

where $\hat{u}_i$ and u_i denote the predicted and ground truth values at the point of the grid *i* (or sensor) and *N* is the total number of spatial–temporal samples in the test set. Here, MAE was selected for its direct physical interpretability. Since our quantities of interest all have physical units, MAE provides the average point-wise deviation in the same units, making it convenient to assess the significance of the reconstruction errors and justify model performance.

Spectral bias measurement: Following the formalism of Kiessling and Thor [30], we adopted a discrete Fourier transform—based formulation to compute the *spectral bias* (SB) and the *fraction of variance unexplained* (FVU) from the reconstruction residuals. Given the ground-truth flow field u and the predicted field $\hat{\mathrm{u}}$, we define the reconstruction residual as

$$\mathrm{r} = \mathrm{u} - \hat{\mathrm{u}}. \tag{12}$$

Following Kiessling and Thor [30], the analysis can incorporate a spatial sampling density function $p(x)$ to account for nonuniform grids. Each field is therefore expressed in its density-weighted form as

$$\mathrm{u}_p(x) = \sqrt{p(x)}\,(\mathrm{u}(x) - \bar{\mathrm{u}}), \qquad \mathrm{r}_p(x) = \sqrt{p(x)}\,(\mathrm{r}(x) - \bar{\mathrm{r}}), \tag{13}$$

where $\bar{\mathrm{u}}$ and $\bar{\mathrm{r}}$ denote the spatial means of u and r, respectively. In our case, $p(x)$ is uniform because the simulation domain is sampled on a regular Cartesian grid.

A two-dimensional discrete Fourier transform (FFT) is then applied to obtain the corresponding spectra:

$$\mathscr{F}_p(\boldsymbol{\omega}) = \mathscr{F}\{\mathrm{u}_p(x)\}, \qquad \mathscr{R}_p(\boldsymbol{\omega}) = \mathscr{F}\{\mathrm{r}_p(x)\}. \tag{14}$$

The cutoff frequency ω_0 that separates the low- and high-frequency regions is selected so that the cumulative energy of $\mathscr{F}_p$ below ω_0 equals that above it, effectively dividing the target spectrum into two halves of equal variance. Using this cutoff, the residual energies are defined as:

$$E_{\text{low}} = \frac{\sum_{\|\boldsymbol{\omega}\|_\infty \le \omega_0} |\mathscr{R}_p(\boldsymbol{\omega})|^2}{\sum_{\boldsymbol{\omega}} |\mathscr{F}_p(\boldsymbol{\omega})|^2}; \qquad E_{\text{high}} = \frac{\sum_{\|\boldsymbol{\omega}\|_\infty > \omega_0} |\mathscr{R}_p(\boldsymbol{\omega})|^2}{\sum_{\boldsymbol{\omega}} |\mathscr{F}_p(\boldsymbol{\omega})|^2}. \tag{15}$$

The general reconstruction error, or *the unexplained fraction of variance* (FVU), is defined as the sum of normalized low- and high-frequency errors:

$$\text{FVU} = E_{\text{low}} + E_{\text{high}}. \tag{16}$$

Finally, the spectral bias is computed as:

$$\text{SB} = \frac{E_{\text{high}} - E_{\text{low}}}{E_{\text{high}} + E_{\text{low}}} \tag{17}$$

where positive values ($\text{SB} > 0$) indicate that residual energy is concentrated in the high-frequency range, implying that the model reconstructs low-frequency components more accurately and thus exhibits a low-

frequency bias. An SB value near zero corresponds to a spectrally balanced reconstruction, while larger magnitudes indicate stronger imbalance between low- and high-frequency errors.

Spectral energy ratio: To evaluate the spectral fidelity of reconstructed flow fields, we define the spectral energy ratio metric, which quantifies the agreement between the predicted and ground-truth energy distributions across spatial wavenumbers. For each representative field snapshot, we compute the two-dimensional discrete Fourier transform (FFT) of both the ground-truth velocity field u and the predicted field û. The FFT provides the complex amplitude distribution as a function of the spatial wavenumber vector $\mathrm{k} = (k_x, k_y)$. The corresponding power spectra are obtained from the squared magnitudes of the FFT coefficients:

$$E_{\mathrm{GT}}(\mathrm{k}) = \left|\mathscr{F}\{\mathrm{u}\}(\mathrm{k})\right|^2, \qquad E_{\mathrm{model}}(\mathrm{k}) = \left|\mathscr{F}\{\hat{\mathrm{u}}\}(\mathrm{k})\right|^2. \tag{18}$$

The isotropic energy distributions $E_{\mathrm{GT}}(k)$ and $E_{\mathrm{model}}(k)$ are obtained by radial averaging over wavenumber shells of equal magnitude $k = \|\mathrm{k}\|$. The spectral energy ratio is then defined as $\frac{E_{\mathrm{model}}(k)}{E_{\mathrm{GT}}(k)}$. A value of 1 indicates perfect agreement between the prediction and the reference in the corresponding wavenumber. Values greater than 1 indicate an overestimation of spectral energy (excess amplitude) on that scale, while values smaller than 1 indicate an underestimation (energy loss). This metric reveals how well each model preserves the energy distribution across different scales.

Result and discussion

In this section, we discuss the validation results of FLRNet. For benchmarking purposes, we compare the field reconstruction performance of FLRNet and its variants with several baseline methods, including the multi-layered perceptron network (MLP), an upgraded version of the shallow network by Erichson et al.[9], and the POD-based method[6]. These methods were re-implemented strictly following the guidelines provided by the authors. In addition, to ensure a fair comparison, we tuned the architecture of the MLP and POD-based models so that they have the same or larger model size compared to FLRNet. In the case of the POD-based method, the number of POD modes was selected as 256, equivalent to the latent size of FLRNet's VAE. For FLRNet, we examined four different variants: FLRNet with vanilla VAE (FLRNet-Std), FLRNet with perceptual loss only (FLRNet-P), FLRNet with Fourier feature only (FLRNet-F), and FLRNet with both Fourier feature and perceptual loss (FLRNet-FP). Table 2 summarizes all the configurations of models used for experiments.

Qualitative assessment

Figures 3–6 show the reconstructed flow fields by FLRNet and other baselines for the flow case with $Re = 60, 300, 950, 3000$, respectively. Those cases represent four distinct flow regimes: steady, symmetric laminar flow ($Re = 60$), periodic vortex shedding ($Re = 300$), quasi-periodic vortex shedding ($Re = 950$) and highly unsteady and chaotic wake flow ($Re = 3000$).

Within the steady laminar flow regime (cf. Fig. 3), all methods demonstrate comparable accuracy in field reconstruction, showing only minor differences in the reconstructed wake structures compared to the ground truth, except at the early stage of the flow (i.e., $t = 0.05$ s). At this stage, while the ground truth exhibits smooth, steady gradients around the obstacle, the MLP and POD methods tend to oversmooth the near-obstacle region and fail to capture the emerging flow. Meanwhile, the FLRNet variants, although better at reproducing the flow behind the obstacle, still exhibit slight deviations from the ground truth.

When the flow becomes more complex with vortex shedding features ($Re = 300$ and $Re = 950$, see Figs. 4 and 5), the performance differences become more prominent. In these more complex flows, FLRNet variants consistently maintain better flow reconstruction quality, with vorticity structures and wake boundaries better captured compared to other baselines. Notably, FLRNet with both Fourier features and the use of perceptual loss during training can preserve the sharp flow gradients and coherent vortex cores. Meanwhile, reconstructions by both POD and MLP exhibit excessive smoothing that blurs the essential features of the flow field. In particular, MLP reconstructions almost miss the flow features in the middle of the flow (at $t = 1.02$ s).

The performance distinction between FLRNet and other baselines is most clearly shown in cases of chaotic flow ($Re = 3000$, see Fig. 6). In this highly complex flow regime, while FLRNet variants maintain their performance in capturing large-scale chaotic flow features, other baselines exhibit a significant performance drop. Although MLP cannot fully reconstruct the chaotic flow features, the reconstruction field by POD is overly diffused and significantly underpredicts the flow velocity magnitude. However, resolving fine-scale details of chaotic flows remains difficult for FLRNet, as its reconstruction often averages out predictions in areas where these fine-scale features matter.

Method	Representation learning (Nb. of parameters)	Latent dimension/ (Nb. of modes)	Sensor encoding (Nb. of parameters)	Model size (Parameters)
FLRNet	VAE (∼ 3.2 M)	256	Attention (∼ 500 K)	∼ 3.7 M
MLP baseline	N/A	N/A	MLP (∼ 8.4 M)	∼ 8.4 M
POD-based model	POD	256 POD modes	MLP (∼ 600 K)	∼ 600 K

Table 2. Model configurations and architectural components used in experiments. All baseline models were re-implemented following the original authors' guidelines with comparable model capacity.

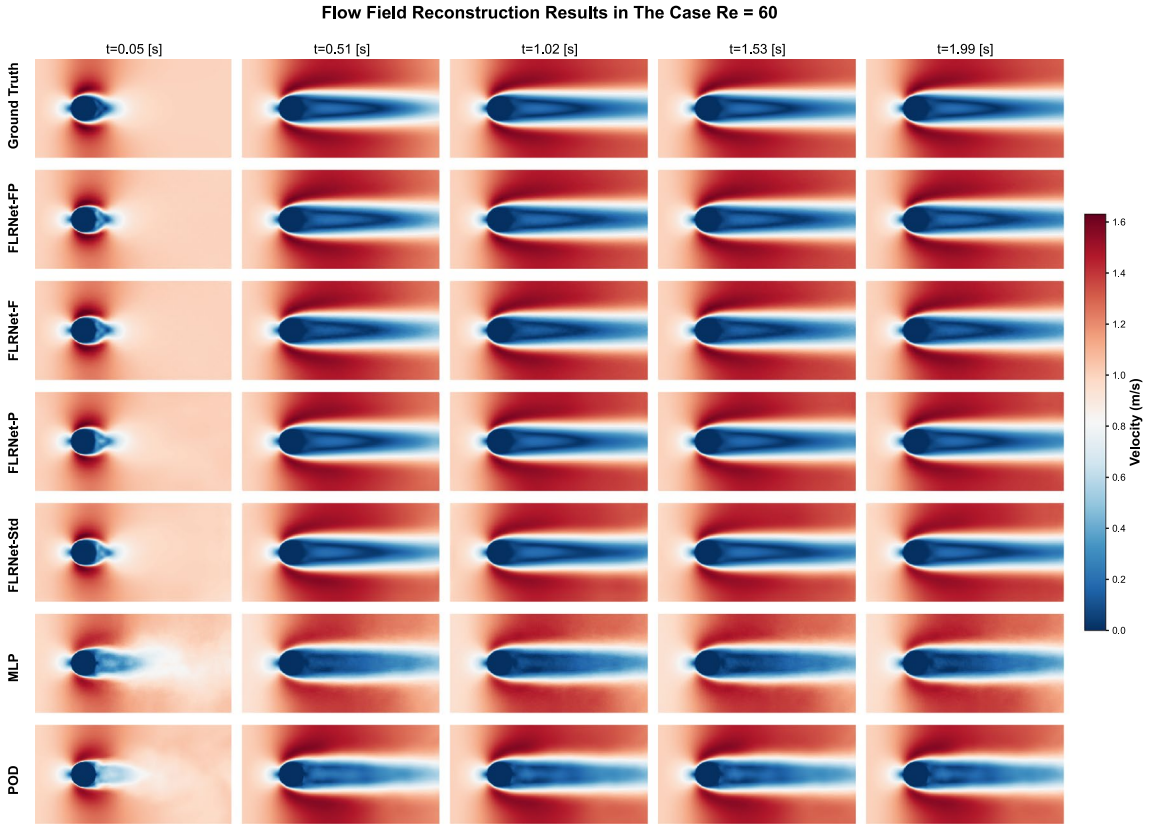


Fig. 3. Reconstructed flow fields by FLRNet and baseline methods at different simulation times for the flow case with $Re = 60$ (steady, symmetric laminar flow) and random sensor layout with 32 sensors used.

Among all FLRNet variants, the FLRNet-FP provides the most accurate and detailed flow reconstruction in all test cases. This suggests that the use of Fourier feature and perceptual loss allow the model to capture more detail flow physics leading to highly accurate and detailed flow reconstruction. However, validation across different flow regimes also showed that the performance gain is only noticeable when the complexity of flow increases. For example, in the case of $Re = 60$ (cf. Fig. 3), a flow in the steady regime, the difference in reconstruction performance is not recognizable.

We continued to analyze MAE profiles across vertical cross sections at six different stream locations to further assess reconstruction performance. Figure 7 presents these error profiles, which were examined at the horizontal positions indicated in Fig. 7a. The results show a strong spatial dependence of the reconstruction error. The reconstruction error remains low in the upstream region (horizontal coordinate of 0.2812 m), increases significantly near the obstacle (horizontal coordinates of 0.5703 m and 0.8516 m), then decreases and stabilizes in the downstream region (horizontal coordinates of 1.1406 m, 1.4219 m, and 1.7109 m). The highest errors occur in the region behind the obstacle, where MAE peaks reach approximately 0.3 m/s. These regions are the locations where flow instabilities develop and vortex shedding begins in intermediate and high Reynolds number flows, highlighting the inherent difficulty in accurately reconstructing complex flow phenomena.

In terms of method performance, MLP exhibits the highest MAE across the cross sections. Meanwhile, POD demonstrates better performance than MLP; however, it still shows significantly higher MAEs compared to FLRNet reconstructions in several regions. Specifically, the MAEs of POD's reconstructions are approximately double those of FLRNet in the downstream region. In the case of FLRNet and its variants, FLRNet (F+P) is consistently the most accurate model, with reconstructions having the lowest MAE at all cross sections. Although other FLRNet variants yield comparable performance, their reconstruction MAEs remain slightly higher than those of FLRNet-FP.

Reconstruction accuracy

The qualitative assessments above are supported by the quantitative analysis presented in Table 3, which reports the reconstruction accuracy of FLRNet and other baselines in various sensor configuration settings. As can be seen, the results demonstrate FLRNet's superior reconstruction capability compared to other baselines. In particular, all FLRNet variants consistently yield better performance compared to MLP and POD, achieving lower MAE reconstructions across all sensor configurations. For example, in the case of 8 sensors used, an extremely sparse measurement scenario, FLRNet-FP achieves an MAE of just 0.068 m/s, approximately two-thirds that of POD and half that of MLP. As the number of sensors increases, providing richer input information for all models, FLRNet's performance superiority persists with an even larger performance gap. With 32 sensors,

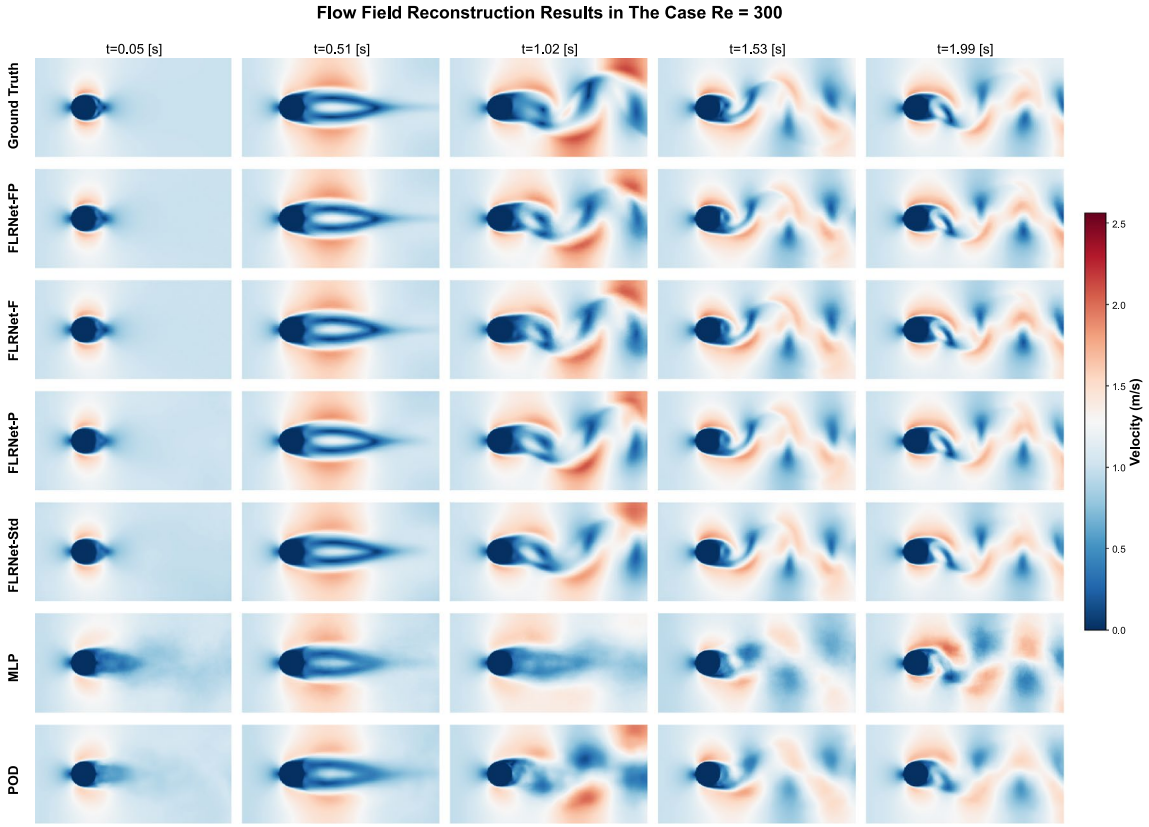


Fig. 4. Reconstructed flow fields by FLRNet and baseline methods at different simulation times for the flow case with $Re = 300$ (periodic vortex shedding) and random sensor layout with 32 sensors used.

FLRNet-FP achieves an error that is only 60% of POD's and 35% of MLP's. This scaling behavior suggests that FLRNet's architecture and training mechanism better leverage additional measurement information compared to other baselines.

The validation results with different sensor layouts also reveal several interesting observations. All methods experience increased difficulty when sensors are positioned around obstacles, with FLRNet and POD showing comparable performance (MAE approximately 0.094 m/s) while the reconstruction error by MLP remains high (MAE of 0.144 m/s). Although FLRNet-FP is still the most accurate model, the performance gap between methods is less pronounced in this challenging configuration. In the case where sensors are distributed near the edge of the domain, FLRNet achieves performance comparable to random sensor distribution, with an MAE of 0.050 m/s for FLRNet-FP. Meanwhile, POD and MLP show only slight improvements in their performance in the sensors around the obstacles configuration, with MAE of 0.083 m/s and 0.115 m/s, respectively. This analysis reveals that FLRNet consistently outperforms other baselines across all sensor configurations, despite some challenges when sensors are placed around obstacles.

The comparison between FLRNet variants also reveals some important insights. From the results presented in Table 3, it is evident that Fourier features have a significant impact on model performance, especially in the extremely sparse sensors case (8-sensor setting). In particular, in this setting, adding Fourier features to the architecture reduces the reconstruction error by approximately 17%. Regarding the addition of perceptual loss, although its impact is less significant than Fourier features, it also contributes positively to the enhanced performance of FLRNet. Without Fourier features, perceptual loss reduces the FLRNet reconstruction error by approximately 6%; while the combination of Fourier features and perceptual loss enhances performance by 18%. This consistent trend of error reduction across all sensor settings demonstrates the positive effects of incorporating both Fourier features and perceptual loss into FLRNet's VAE architecture.

Temporal evolution of reconstruction error

We examine the temporal progression of the reconstruction error associated with FLRNet and other baselines in time throughout the simulation to assess model performance at various temporal phases. Figure 8 illustrates the variation in MAE of the predictions of FLRNet and other baselines as the simulation progresses. During the initial phase of the simulation ($t = 0$–0.4 s), all methodologies demonstrate consistently low and stable errors. This phase corresponds to the initialization stage when the flow begins to enter the domain. During this period, only simplistic flow patterns are established around the obstacle without any appearance of instability; therefore, reconstructing the flow reconstruction during this time period is relatively straightforward and all models perform proficiently.

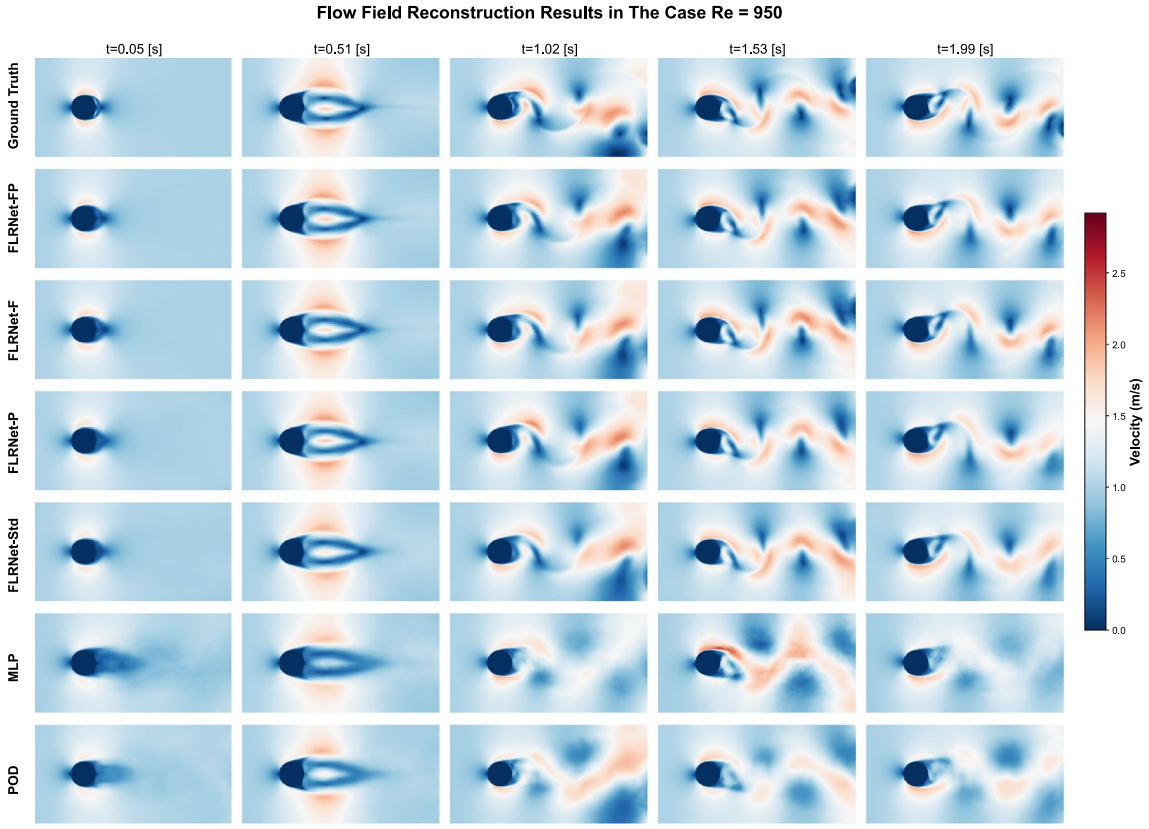


Fig. 5. Reconstructed flow fields by FLRNet and baseline methods at different simulation times for the flow case with $Re = 950$ (quasi-periodic vortex shedding) and random sensor layout with 32 sensors used.

During the transition phase ($t = 0.4$–0.8 s), the flow evolves further, leading to the appearance of instability. In this phase, the reconstruction error increases significantly for all methods. This phase represents the most challenging period for accurately reconstructing the flow field, owing to the development of transient vortex formations and periodic shedding patterns that exhibit high dynamism and are extremely challenging to capture accurately from sparse sensor data. Upon complete development of periodic vortex shedding, the reconstruction error stabilizes and begins to diminish as the flow achieves greater stability. Throughout this period, FLRNet variants exhibit exceptional consistency, maintaining a mean absolute error (MAE) of approximately 0.05 m/s with minimal temporal fluctuation. In contrast, MLP and POD persistently exhibit significantly higher errors of approximately 0.14 m/s and 0.08 m/s, respectively, compared to FLRNet.

Throughout this process, FLRNet consistently achieves the lowest MAE among all baselines, demonstrating reliable reconstruction performance at all stages of the simulation. Both the Fourier and Fourier-perceptual variants exhibit nearly identical temporal behavior, indicating that spectral feature encoding enhances stability and consistency in reconstruction quality. In contrast, the MLP and POD methods produce larger errors throughout the simulation.

Model performance under addition of noise perturbation

FLRNet is trained on a simulation, clean data set of fluid flow; however, the real-world sensor measurement in practice always contains noise. Therefore, we validate the robustness of FLRNet by assessing their prediction accuracy at different levels of noise added to the sensor measurement. In this experiment, we assume that the sensor measurements of the flow field follow the Gaussian distribution.

$$\mathrm{y}_{\mathrm{noise}} = \mathrm{y} + \mathcal{N}(0, \sigma^2). \tag{19}$$

Consequently, we vary the added perturbations from 0% to 20% and compute the reconstruction error. Figure 9 illustrates the noise resilience capacity of FLRNet compared to other baselines. At low noise levels (added noise below 5%), FLRNet maintains its superior performance compared to other baselines with a reconstruction error below 0.05 m/s, while POD has an MAE of approximately 0.07 m/s and that of MLP is above 0.1 m/s. The reconstruction error escalates significantly for all models when the noise increases to more than 5% up to 20%, with MAE above 0.1 m/s for all models. In particular, across all levels of added noise, the MAE of the FLRNet reconstruction is consistently lower compared to MLP and POD, indicating its robust performance even in the presence of noisy input. This performance robustness makes FLRNet suitable for real-world implementations where noise perturbations can make sensor measurements imprecise.

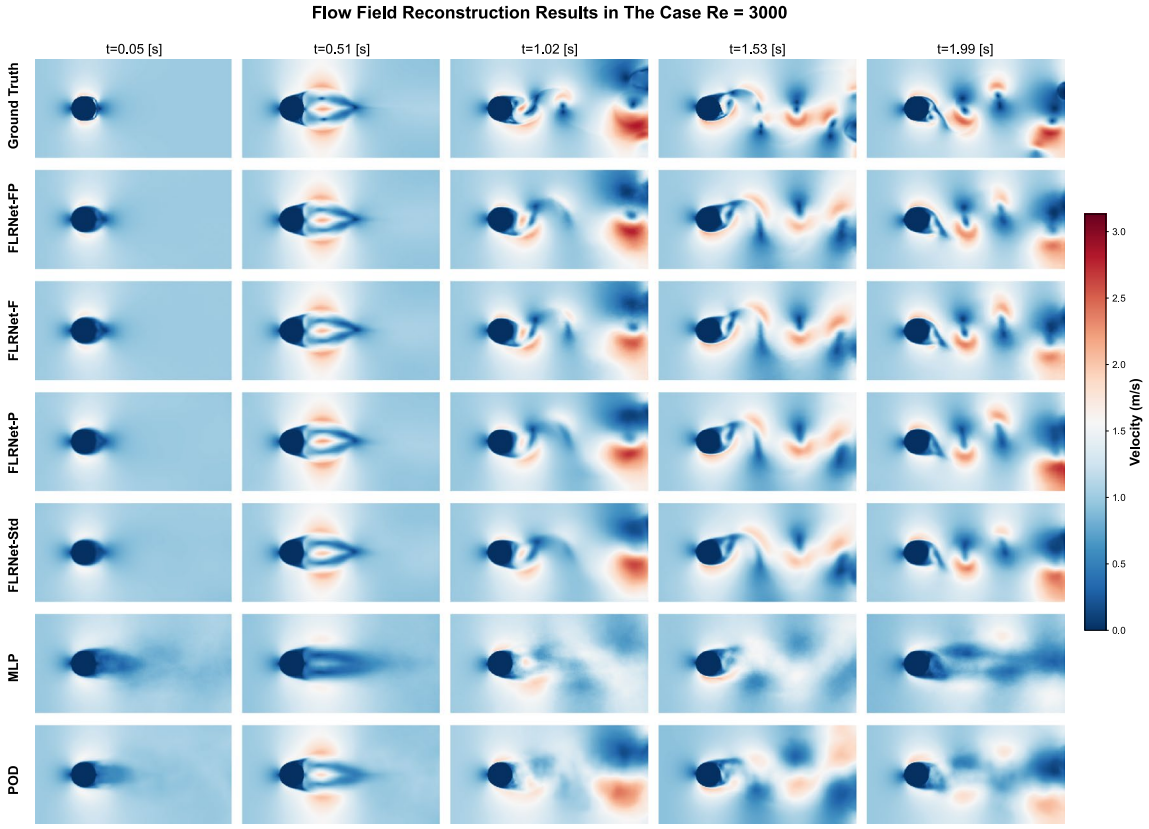


Fig. 6. Reconstructed flow fields by FLRNet and baseline methods at different simulation times for the flow case with $Re = 3000$ (highly unsteady and chaotic wake flow) and random sensor layout with 32 sensors used.

Performance consistency across different flow regimes

The ultimate benefit of FLRNet lies in its reliable performance across different flow regimes. Figure 10 presents the MAE variation of FLRNet in relation to other baseline models over a wide range of Reynolds numbers, from $Re = 30$ to 8000. In both steady and weakly unsteady regimes ($Re = 30$–300), FLRNet variants exhibit MAE below 0.05 m/s, with only a slight increase in MAE as *Re* increases from 30 to 300. In contrast, the reconstruction errors of other baselines are at least twice that of FLRNet (POD), or they experience significant increases in errors (MLP). As the flow begins to transition from steady to periodic vortex shedding, FLRNet consistently performs well, with the maximum MAE just exceeding 0.05 m/s at $Re = 950$. Both POD and MLP show a steady drop in reconstruction accuracy, achieving an MAE of approximately 0.08 m/s for POD and exceeding 0.125 m/s for MLP. Upon entering the chaotic region, all models see a similar rate of error increase with increasing Reynolds numbers. However, FLRNet continues to offer the highest prediction quality, highlighting its superior performance over other baselines. Notably, at the highest examined Reynolds number, the average MAE of the FLRNet series remains below 0.1 m/s, whereas POD's is over 0.125 m/s, and MLP's is nearly 0.2 m/s. This Reynolds number scaling analysis has shown that FLRNet variants maintain practical reconstruction accuracy ($\mathrm{MAE} < 0.10$ m/s) throughout the tested range, demonstrating their capability to deal with different flow regimes, while baseline methods become increasingly unreliable as flow complexity increases.

Spectral error analysis

We analyze the spectral error of all models using the metrics described in the 'Validation Metrics' section. Figure 11 reports the SB and FVU values of all baselines across different flow regimes. Overall, the SB values of most models are positive, except in the steady laminar regime ($\mathrm{Re} = 30$ and $\mathrm{Re} = 60$) for certain baselines, and in the chaotic wake regime ($\mathrm{Re} = 8000$) for the MLP model. For $\mathrm{Re} = 30$, while all FLRNet variants exhibit positive SB values, both MLP and POD show negative SB values. The positive SB of FLRNet variants suggests that they capture low-frequency components better than high-frequency ones, that is, E_{low} is relatively smaller than E_{high}. This is not necessarily undesirable, since laminar flows are dominated by smooth structures, making such behavior of FLRNet understandable. In contrast, the negative SB observed for MLP and POD at $\mathrm{Re} = 30$ indicates that their low-frequency residuals are higher than their high-frequency ones, reflecting poor capability to model the dominant smooth features of this regime.

As flow complexity increases, the SB values of all models increase and eventually saturate, with a slight drop observed for FLRNet as Re increases from 30 to 60. At higher Reynolds numbers, FLRNet consistently exhibits higher SB values than both POD and MLP, implying a stronger bias toward low-frequency components. However, this interpretation should be considered together with the FVU results in Fig. 11b. Although the SB values of

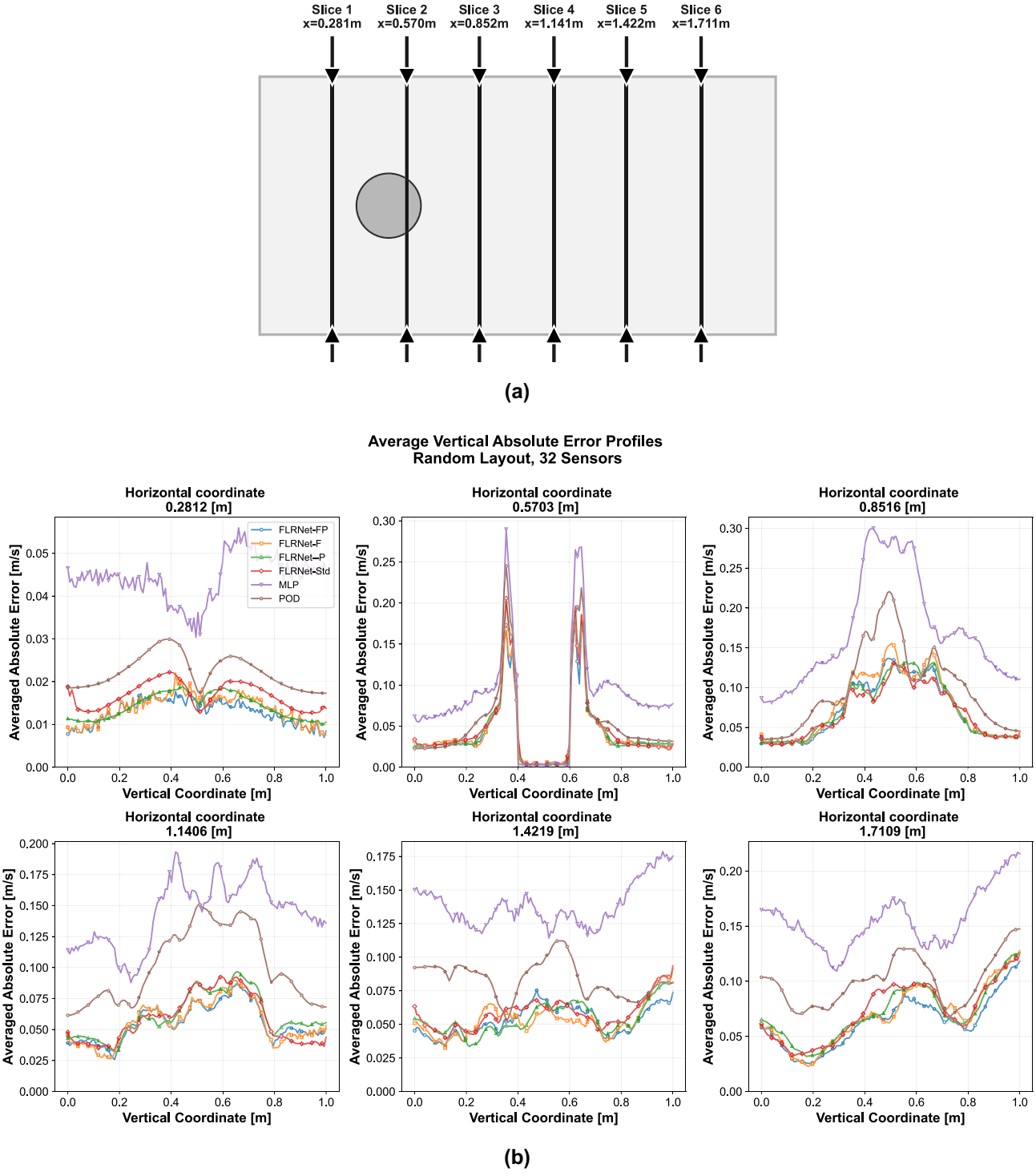


Fig. 7. Reynolds-number-averaged error profiles of FLRNet and baseline models at different cross-sections. (**a**) Horizontal positions of the examined cross-sections. (**b**) Error profiles of all models across the six selected cross-sections.

MLP cluster near zero, indicating seemingly balanced learning behavior, its FVU remains the highest across all regimes, signifying poor reconstruction accuracy. That means a near-zero SB, in combination with high FVU, implies that residuals for both low- and high-frequency components are large, reflecting low modeling capability rather than a true balance learning.

Similarly, POD achieves a somewhat better frequency balance than FLRNet but at the cost of much higher FVU, suggesting that this 'balanced learning' does not translate into overall performance gains. Nevertheless,

Model	Velocity mean absolute error (m/s)				
	Random distribution			Around obstacle	Near edge
	8 sensors	16 sensors	32 sensors	32 sensors	32 sensors
FLRNet-Std	0.083	0.065	0.045	0.098	0.060
FLRNet-P	0.078	0.057	0.043	0.095	0.076
FLRNet-F	0.069	0.053	0.041	0.095	0.051
FLRNet-FP	0.068	0.051	0.039	0.093	0.050
MLP	0.124	0.126	0.110	0.144	0.115
POD	0.085	0.073	0.069	0.094	0.083

Table 3. Average reconstruction accuracy of FLRNet and the other baseline models.

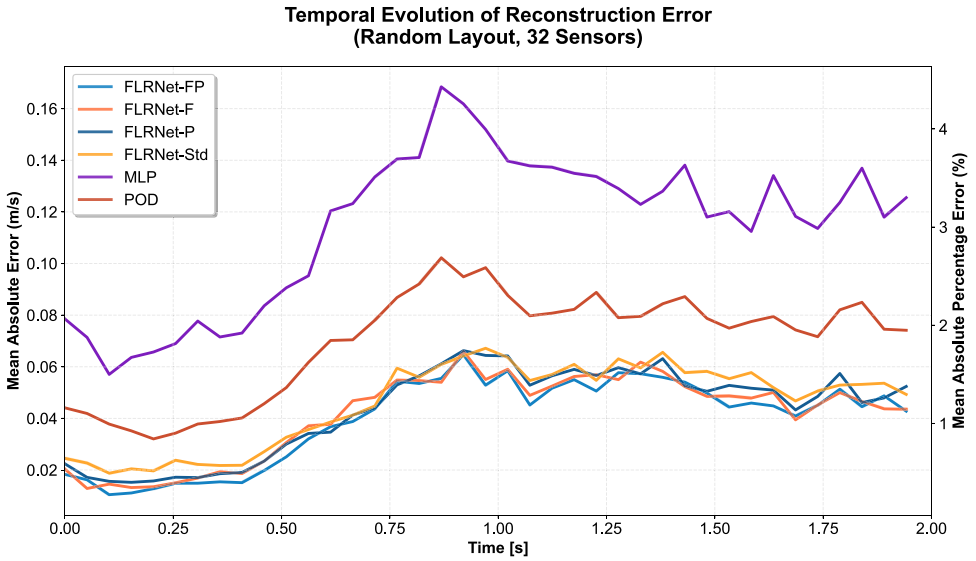


Fig. 8. Analysis of the reconstruction error w.r.t the temporal evolution of the flow field. The reconstruction error increases for all models, peaks at around $t = 0.8$s when the instability arises and gradually reduces as the flow enters its equilibrium. During the whole process, FLRNet with Fourier feature yield the lowest MAE among all baselines.

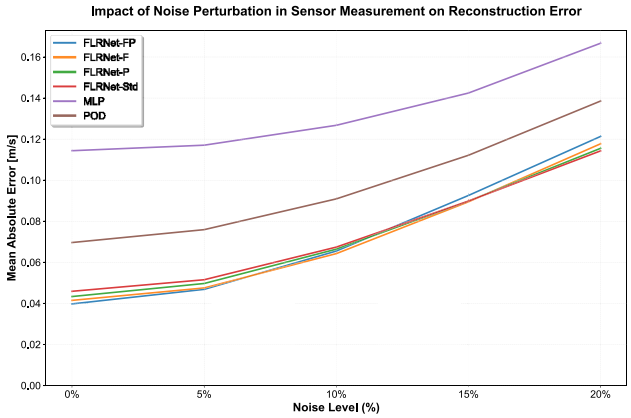


Fig. 9. Effect of noise in sensor measurement. Compared to other baselines, FLRNet is the most robust method as its MAE is the lowest as the level of noise increase.

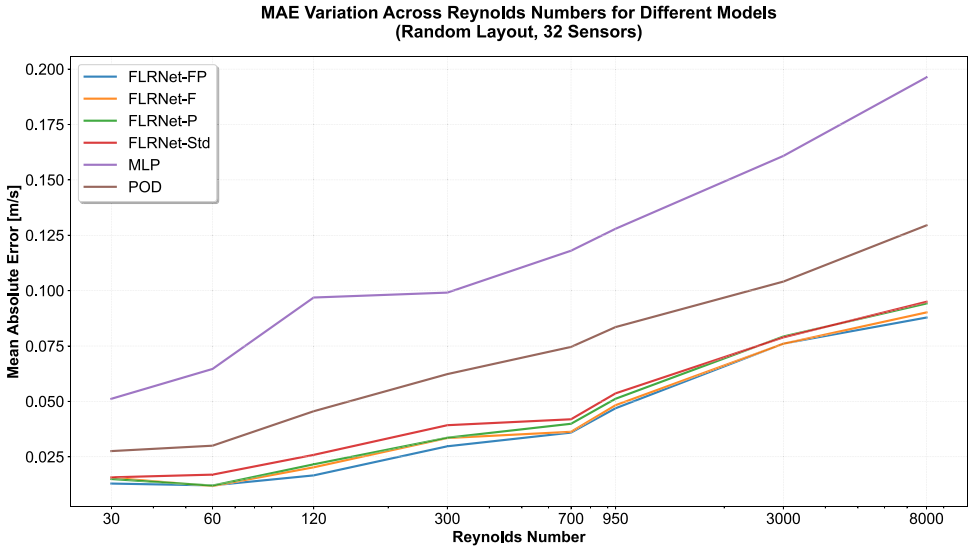


Fig. 10. MAE variation of FLRNet in relation to other baseline models across different Reynolds numbers.

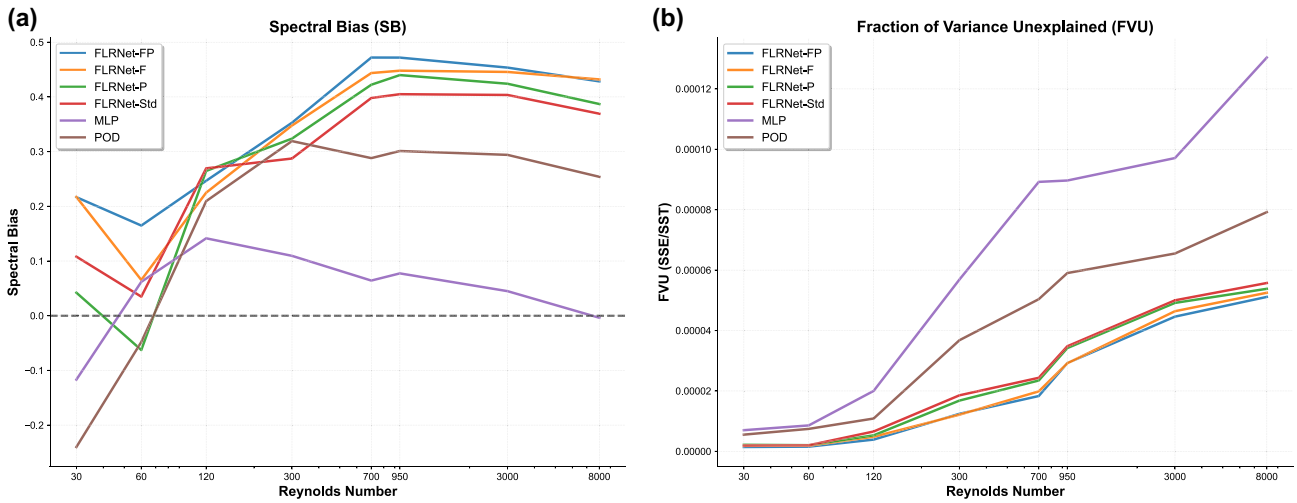


Fig. 11. (**a**) Spectral bias (SB) and (**b**) fraction of variance unexplained (FVU) of all baselines across different flow regimes.

this observation hints at a potential future direction, combining FLRNet with mode-decomposition techniques to achieve both balanced frequency learning and superior reconstruction performance.

Further evidence supporting these observations is provided in Fig. 12, which presents the spectral energy ratio plots, $E_{\mathrm{model}}(k)/E_{\mathrm{GT}}(k)$, across spatial wavenumbers k. These spectra quantify the ability of each model to reproduce the ground-truth energy distribution across spatial scales. As shown, FLRNet-FP maintains better agreement with ground truth ($E_{\mathrm{model}}(k)/E_{\mathrm{GT}}(k)$ closer to 1) over a wide wavenumber range, including the high-wavenumber region where small-scale gradients and fine structures dominate. This indicates that FLRNet accurately preserves the fine-scale energy content while maintaining coherent large-scale features–consistent with its moderate positive SB and low FVU values. In contrast, both MLP and POD progressively underpredict the spectral energy in larger k, revealing a loss of high-frequency structures typical of spectrally biased or underfit models. Together, the SB, FVU, and spectral energy ratio analysis consistently demonstrate that FLRNet not only mitigates spectral bias but also sustains physically meaningful flow reconstructions across a wide range of Reynolds numbers.

Conclusion

We present FLRNet, a DL method, for flow field reconstruction from sparse sensor measurements. FLRNet employs a VAE network to learn a low-dimensional latent representation and an MLP network to correlate the sensor measurement to the learned representation. FLRNet equips its VAE with Fourier feature layers and adds perceptual loss to enhance its ability to learn detailed features of the flow field. Experimental studies

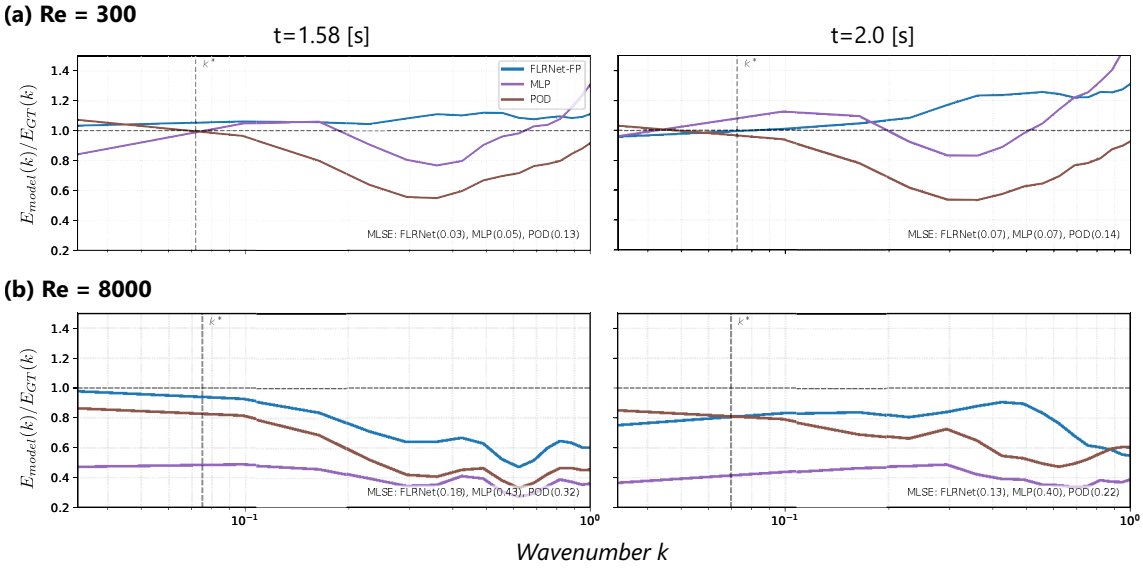


Fig. 12. Visualization of the residual power spectra obtained via FFT, showing the ratio $E_{\mathrm{model}}(k)/E_{\mathrm{GT}}(k)$ for FLRNet-FP, MLP, and POD at representative times $t = 1.58,\ 2.0\,(\mathrm{s})$ for (**a**) $\mathrm{Re} = 300$ and (**b**) $\mathrm{Re} = 8000$. Dashed horizontal lines indicate unity, and the vertical line (k^*) marks the wavenumber at which 50% of the cumulative ground-truth energy is contained. Lower MLSE values denote smaller spectral deviation, highlighting FLRNet-FP's superior ability to capture high-frequency energy components compared to the baseline models.

showed that added Fourier feature layers and perceptual loss help FLRNet overcome the spectral bias issue, improve its reconstruction accuracy, and increase its robustness to noise in sensor measurement compared to other traditional methods relying solely on MLP networks or linear dimension reduction techniques such as POD. Furthermore, the results of the numerical experiment show that FLRNet exhibits consistent superior performance across different flow regimes compared to other baselines, which is the current gap in the literature. Future works will expand the ability of FLRNet to cope with different sensor settings, including variation in the number of sensors used and their positions. We are also working on enabling FLRNet to handle random discretization grid systems in both the spatial and temporal domains under investigation. This is a missed opportunity in the current design of FLRNet. Finally, we are preparing to expand the application of FLRNet to other domains such as material science or experimental fluid dynamics.

Data availability

The implementation details, trained models, source code, and datasets used in this study are available in this code repository: https://github.com/hphong1990/flrnet.git

Received: 26 November 2024; Accepted: 12 January 2026
Published online: 22 January 2026

References

1. Callaham, J. L., Maeda, K. & Brunton, S. L. Robust flow reconstruction from limited measurements via sparse representation. *Phys. Rev. Fluids* **4**, 103907. https://doi.org/10.1103/PhysRevFluids.4.103907 (2019).
2. Rowley, C. W. & Dawson, S. T. Model reduction for flow analysis and control. *Annu. Rev. Fluid Mech.* **49**, 387–417. https://doi.org/10.1146/annurev-fluid-010816-060042 (2017).
3. Wu, J., Xiao, D. & Luo, M. Deep-learning assisted reduced order model for high-dimensional flow prediction from sparse data. *Phys. Fluids* **35**, https://doi.org/10.1063/5.0166114 (2023).
4. Caverly, D. & Nedic, J. Estimating wind velocity and direction using sparse sensors on a cylinder. *Phys. Fluids*. **34**, https://doi.org/10.1063/5.0112658 (2022).
5. Tan, Z., Li, R. & Zhang, Y. Flow field reconstruction of 2d hypersonic inlets based on a variational autoencoder. *Aerospace*. **10**, https://doi.org/10.3390/aerospace10090825 (2023).
6. Dubois, P., Gomez, T., Planckaert, L. & Perret, L. Machine learning for fluid flow reconstruction from limited measurements. *J. Comput. Phys.* **448**, 110733. https://doi.org/10.1016/j.jcp.2021.110733 (2022).
7. Loiseau, J.-C., Noack, B. R. & Brunton, S. L. Sparse reduced-order modelling: sensor-based dynamics to full-state estimation. *J. Fluid Mech.* **844**, 459–490. https://doi.org/10.1017/jfm.2018.147 (2018).
8. Mons, V., Chassaing, J.-C., Gomez, T. & Sagaut, P. Reconstruction of unsteady viscous flows using data assimilation schemes. *J. Comput. Phys.* **316**, 255–280. https://doi.org/10.1016/j.jcp.2016.04.022 (2016).
9. Erichson, N. B. et al. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **476**, 20200097. https://doi.org/10.1098/rspa.2020.0097 (2020).
10. Li, R. et al. Deep learning reconstruction of high-reynolds-number turbulent flow field around a cylinder based on limited sensors. *Ocean Eng.* **304**, 117857. https://doi.org/10.1016/j.oceaneng.2024.117857 (2024).
11. Zhao, X. et al. Recfno: A resolution-invariant flow and heat field reconstruction method from sparse observations via fourier neural operator. *Int. J. Therm. Sci.* **195**, 108619. https://doi.org/10.1016/j.ijthermalsci.2023.108619 (2024).

12. Xie, P. et al. A physics-informed deep learning model to reconstruct turbulent wake from random sparse data. *Phys. Fluids* **36**, 065145. https://doi.org/10.1063/5.0212298 (2024).
13. Wang, S., Wang, H. & Perdikaris, P. On the eigenvector bias of fourier feature networks: From regression to solving multi-scale pdes with physics-informed neural networks. *Comput. Methods Appl. Mech. Eng.* **384**, 113938. https://doi.org/10.1016/j.cma.2021.113938 (2021).
14. Nguyen, P. C. H. et al. Parcv2: Physics-aware recurrent convolutional neural networks for spatiotemporal dynamics modeling. ArXiv:abs/2402.12503 (2024).
15. Taira, K. et al. Modal analysis of fluid flows: An overview. *AIAA J.* **55**, 4013–4041. https://doi.org/10.2514/1.J056060 (2017).
16. Callaham, J. L., Maeda, K. & Brunton, S. L. Robust flow reconstruction from limited measurements via sparse representation. *Phys. Rev. Fluids* **4**, 103907. https://doi.org/10.1103/PhysRevFluids.4.103907 (2019).
17. Noack, B. R., Afanasiev, K., Morzynski, M., Tadmor, G. & Thiele, F. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.* **497**, 335–363. https://doi.org/10.1017/S0022112003006694 (2003).
18. Cheng, X. et al. Physics-aware recurrent convolutional neural networks for modeling multiphase compressible flows. *Int. J. Multiph. Flow* **177**, 104877. https://doi.org/10.1016/j.ijmultiphaseflow.2024.104877 (2024).
19. Nguyen, P.C.H. et al. Parc: Physics-aware recurrent convolutional neural networks to assimilate meso scale reactive mechanics of energetic materials. *Sci. Adv.* **9**, eadd6868. https://doi.org/10.1126/sciadv.add6868 (2023).
20. Arnault, A., Dandois, J. & Foucaut, J.-M. Comparison of stochastic estimation methods with conditional events optimization for the reconstruction of the flow around a supercritical airfoil in transonic conditions. *Comput. Fluids* **136**, 436–455. https://doi.org/10.1016/j.compfluid.2016.06.006 (2016).
21. Peng, X., Li, X., Chen, X., Chen, X. & Yao, W. A hybrid deep learning framework for unsteady periodic flow field reconstruction based on frequency and residual learning. *Aerosp. Sci. Technol.* **141**, 108539. https://doi.org/10.1016/j.ast.2023.108539 (2023).
22. Liu, T., Li, Y., Jing, Q., Xie, Y. & Zhang, D. Supervised learning method for the physical field reconstruction in a nanofluid heat transfer problem. *Int. J. Heat Mass Transf.* **165**, 120684. https://doi.org/10.1016/j.ijheatmasstransfer.2020.120684 (2021).
23. Peng, X., Li, X., Gong, Z., Zhao, X. & Yao, W. A deep learning method based on partition modeling for reconstructing temperature field. *Int. J. Therm. Sci.* **182**, 107802. https://doi.org/10.1016/j.ijthermalsci.2022.107802 (2022).
24. Kingma, D. P. & Welling, M. An introduction to variational autoencoders. *CoRR* arxiv:abs/1906.02691 (2019).
25. Rahimi, A. & Recht, B. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, vol. 20 (eds. Platt, J. et al.) (Curran Associates, Inc., 2007).
26. Tancik, M. et al. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*. (2020).
27. Johnson, J., Alahi, A. & Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision—ECCV 2016* (eds. Leibe, B. et al.) 694–711 (Springer International Publishing, 2016).
28. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. Rethinking the inception architecture for computer vision. *CoRR* arxiv:abs/1512.00567 (2015).
29. Deng, J. et al. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. https://doi.org/10.1109/CVPR.2009.5206848 (2009).
30. Kiessling, J. & Thor, F. A computable definition of the spectral bias. *Proc. AAAI Conf. Artif. Intell.* **36**, 7168–7175. https://doi.org/10.1609/aaai.v36i7.20677 (2022).

Author contributions

P.N. conceptualized the method under discussion with J.C. and Q-T.L. P.N implemented the method with J.C and Q-T.L support. P.N and J.C. conducted the experiments and Q-T.L supported the creation of figures. All authors contributed to the writing and review of the manuscripts.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to P.C.H.N.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.