Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

HADF: a hash-adaptive dual fusion implicit network for super-resolution of turbulent flows*

Yunfei LIU†1,2,3, Xinhai CHEN†‡1,2,3, Gen ZHANG1,2,3, Qingyang ZHANG1,2,3, Qinglin WANG1,2,3, Jie LIU1,2,3

[1]*National Key Laboratory of Parallel and Distributed Computing, National University of Defense Technology, Changsha 410073, China*

[2]*Laboratory of Digitizing Software for Frontier Equipment, National University of Defense Technology, Changsha 410073, China*

[3]*College of Computer Science and Technology, National University of Defense Technology, Changsha 410073, China*

†E-mail: liuyunfei@nudt.edu.cn; chenxinhai16@nudt.edu.cn

Received June 17, 2025; Revision accepted Oct. 24, 2025; Crosschecked Nov. 9, 2025; Published online Dec. 8, 2025

Abstract: Turbulence, a complex multi-scale phenomenon inherent in fluid flow systems, presents critical challenges and opportunities for understanding physical mechanisms across scientific and engineering domains. Although high-resolution (HR) turbulence data remain indispensable for advancing both theoretical insights and engineering solutions, their acquisition is severely limited by prohibitively high computational costs. While deep learning architectures show transformative potential in reconstructing high-fidelity flow representations from sparse measurements, current methodologies suffer from two inherent constraints: strict reliance on perfectly paired training data and inability to perform multi-scale reconstruction within a unified framework. To address these challenges, we propose HADF, a hash-adaptive dynamic fusion implicit network for turbulence reconstruction. Specifically, we develop a low-resolution (LR) consistency loss that facilitates effective model training under conditions of missing paired data, eliminating the conventional requirement for fully matched LR and HR datasets. We further employ hash-adaptive spatial encoding and dynamic feature fusion to extract turbulence features, mapping them with implicit neural representations for reconstruction at arbitrary resolutions. Experimental results demonstrate that HADF achieves superior performance in global reconstruction accuracy and local physical properties compared to state-of-the-art models. It precisely recovers fine turbulence details for partially unpaired data conditions and diverse resolutions by training only once while maintaining robustness against noise.

Key words: Turbulence reconstruction; Deep learning; Unpaired data; Low-resolution consistency loss; Hash-adaptive spatial encoding; Dynamic feature fusion; Implicit neural representations

https://doi.org/10.1631/FITEE.2500419 **CLC number:** TP391.4; O35

1 Introduction

Turbulence is present in diverse engineering and natural systems, including aerospace propulsion, combustion engines, wind energy infrastructure, and atmospheric dynamics (Davidson, 2015). The inherent nonlinearity and multiscale characteristics of turbulent flows significantly influence system performance and stability (Wang et al., 2022). A comprehensive understanding of turbulent processes is therefore critical for optimizing system design

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 12402349), the Natural Science Foundation of Hunan Province (No. 2024JJ6468), the Youth Foundation of the National University of Defense Technology (No. ZK2023-11), and the National Key Research and Development Program of China (No. 2021YFB0300101)

ORCID: Yunfei LIU, https://orcid.org/0009-0004-3873-2220
Xinhai CHEN, https://orcid.org/0000-0002-2931-4893

and driving technological innovation. To resolve the intricate details of turbulence, acquiring high-resolution (HR) data has become imperative which provides the foundation for accurate characterization and modeling of these dynamic phenomena.

Turbulence data can be directly obtained through professional measurement techniques, such as tomography (Pareja et al., 2019) and laser diagnostics (Liu N and Ma, 2020). These approaches are constrained by instrument precision and experimental complexity, preventing them from fully capturing the HR details of turbulent flows (Liu HC et al., 2019). Alternatively, scientific computing approaches have been developed to simulate complex turbulent flows. These methods resolve the physical conservation equations over fine-grained computational grids to capture HR turbulence structures (Chen XH et al., 2021b). Nevertheless, they typically require complex mesh generation and expensive numerical iterations, leading to substantial computational costs (Cant et al., 2022; Chen XH et al., 2024).

Super-resolution reconstruction provides an optional strategy to address the limitations by recovering HR turbulent flows from low-resolution (LR) inputs (Fukami et al., 2023). Traditional methods, such as interpolation, are commonly employed to generate HR outputs by estimating intermediate values from surrounding LR data. However, because these methods rely only on simplistic data calculations without accounting for relevant physical constraints, they fail to reconstruct the nonlinear turbulent features governed by the Navier–Stokes equations. Recently, deep learning techniques have been employed to extract features from LR data and upsample these latent features to reconstruct HR flow fields (Fukami et al., 2019). By capturing the complex and nonlinear characteristics of turbulent flows, deep learning methods enable accurate reconstruction of intricate turbulence details. Although deep learning methods enhance the reconstruction of turbulence details over traditional interpolation, they still exhibit some limitations. First, existing models typically rely on paired LR and HR data for supervised training. However, in practical turbulence experiments or simulations, unpaired data frequently appear, where LR data lack corresponding HR counterparts. Such unpaired data cannot be used in model training. Because this reduces the amount of available training data and affects model performance, it is important to develop methods that can effectively use unpaired data. Second, existing models are designed for a fixed upsampling ratio and are unable to support multiple resolution tasks within a single model. This limitation renders them inflexible for applications requiring diverse resolution outputs.

To overcome these limitations we propose HADF, a novel network model that reconstructs turbulent flows at arbitrary resolutions with only one training while accurately recovering fine details from partially available HR data. Our network model introduces a hash-adaptive spatial encoding strategy that employs multi-resolution hash encoding to dynamically prioritize critical turbulent regions, efficiently capturing intricate details across diverse flow domains. Additionally, we develop a dynamic feature fusion mechanism with a dual-path architecture that actively integrates content and spatial features, optimizing feature extraction and enhancing reconstruction accuracy. To address the challenge of unpaired data, we develop an LR consistency loss that aligns downsampled HR outputs with the original LR inputs, enabling robust training under partially unpaired data conditions. Experiments on diverse turbulent datasets demonstrate HADF's superior global accuracy and local physical fidelity over state-of-the-art models in reconstructing turbulence structures.

2 Related works

Super-resolution reconstruction is essential for obtaining HR turbulence data, overcoming the limitations of direct measurement techniques constrained by instrument precision and experimental complexity. Additionally, it addresses the substantial computational costs of traditional scientific computing methods, which rely on complex mesh generation and expensive numerical iterations (Chen XH et al., 2021a).

Conventional interpolation methods, which estimate intermediate values by averaging or fitting functions to adjacent LR data points, often produce overly smoothed outputs that fail to capture the intricate small-scale structures of turbulent flows, as shown in Fig. 1.

In light of these challenges, deep learning methods provide a promising alternative by leveraging their ability to model complex, nonlinear turbulence

patterns, enabling HR reconstructions with superior accuracy. Trained on ground truth (GT) HR data, as illustrated in Fig. 2, these approaches extract latent features from LR turbulence data and upsample them through convolutional or generative networks, effectively reconstructing HR turbulent flows (Brunton et al., 2020). In contrast to conventional interpolation methods, which often produce smoothed outputs, deep learning successfully captures intricate small-scale turbulence structures, producing details that traditional methods fail to reconstruct.

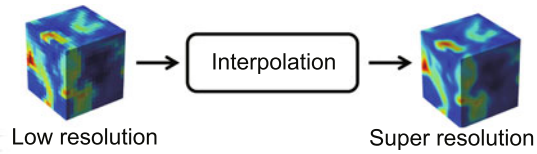

Fig. 1 Architecture of conventional interpolation

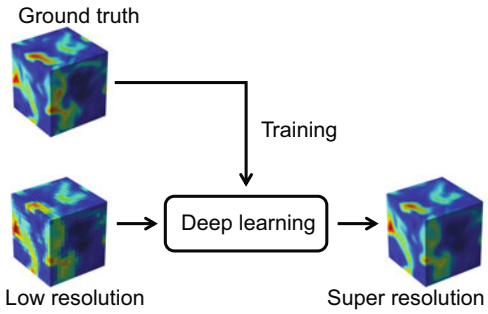

Fig. 2 Architecture of deep learning

Fully connected networks were first employed to reconstruct two-dimensional (2D) velocity fields from sparse sensor measurements. Erichson et al. (2020) adopted fully connected networks to reconstruct turbulent wakes from limited data, successfully recovering vorticity at moderate Reynolds numbers. However, fully connected networks are constrained by computational inefficiency and a limited capacity to capture spatial correlations in turbulent fields. Fukami et al. (2019) applied convolutional neural networks (CNNs) to super-resolution for isotropic turbulence, showing that convolutions outperformed bicubic interpolation in capturing turbulent structures. In addition, Morimoto et al. (2021) proposed specialized padding operations to handle boundary conditions in filtering, achieving CNN architectures suitable for fluid flow analysis. Page (2025) introduced dedicated layer structures to enhance the physically valid reconstruction of turbulent fields. These approaches provide new directions for improving our handling of flow fields.

To address complex three-dimensional (3D) flows, Transformer architectures enable non-local modeling of turbulent flows. Xu Q et al. (2023) developed a Transformer method for super-resolution, leveraging a wider sensing domain to extract deep features and reconstruct high-fidelity turbulent flow fields. To address volumetric coupling in flows, Yang et al. (2023) introduced a back-projection network that iteratively maps multiscale LR features to HR velocity fields, accurately reconstructing vortex structures and kinetic energy spectra in isotropic turbulence and boundary layers. For reactive flows, Xu WJ et al. (2020) developed generative adversarial networks (GANs) that improved prediction sharpness through adversarial training, achieving higher fidelity in turbulent flow fields. Similarly, Bode et al. (2021) implemented a physics-informed GAN for subfilter modeling in reactive flows, combining adversarial training with continuity constraints to improve extrapolation in combustion scenarios.

Despite the remarkable progress of deep learning methods in reconstructing turbulent flow fields, several limitations remain. Current models are constrained to fixed upsampling ratios, lacking the ability to handle multiscale reconstruction tasks within a single framework. Moreover, these models typically rely on supervised training with paired LR and HR data, but acquiring HR data is often prohibitively expensive in many practical scenarios. This dependency limits their reconstruction capability in scenarios where HR data are unavailable. Therefore, it is essential to explore more flexible and data-efficient approaches, with the ability to generalize across scales and remain effective in the absence of HR data.

Implicit neural representations leverage neural networks to map spatial coordinates to physical quantities, representing complex physical fields as continuous functions without reliance on traditional grids (Pan et al., 2023). As demonstrated in Fig. 3, this approach enables arbitrary resolution reconstructions, which accommodate multiscale phenomena and reduce storage demands compared to discrete representations (Shen et al., 2024). However, reconstructing turbulent flow fields with implicit neural representations is challenging due to

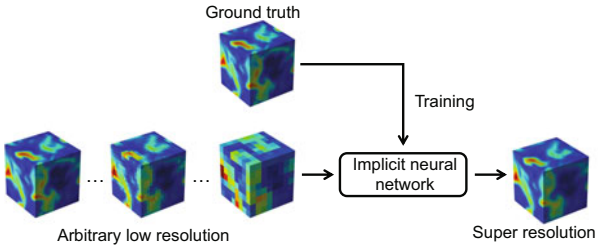

Fig. 3 Architecture of implicit neural representations

turbulence's inherent nonlinear complexity and multiscale nature, particularly in effective feature extraction and multiscale feature integration. Additionally, vanilla implicit neural representation models still rely on paired LR and HR data for training, limiting their reconstruction accuracy when HR samples are unavailable. Although recent studies (Fukami and Taira, 2024; Page, 2025) have demonstrated that it is possible to push the LR limit by leveraging turbulence self-similarity or introducing differentiable solvers, these approaches still face challenges because the snapshot-based method is sensitive to the choice of training regions whereas the solver-based method requires high computational cost and careful treatment of boundary conditions. In this paper, we propose HADF, which reconstructs turbulent flows at arbitrary resolutions after training once. By introducing LR consistency loss, HADF enables robust training with partially unpaired data. Unlike snapshot- or solver-based strategies that either suffer from sensitivity to training regions or incur significant computational overhead, the proposed loss provides a simpler and more efficient mechanism to leverage unpaired data without introducing additional solver complexity. Additionally, its hash-adaptive spatial encoding and dynamic feature fusion mechanisms enhance feature extraction and integration, enabling precise reconstruction of turbulence details.

3 Methodology

In this section, we introduce our proposed network model. As illustrated in Fig. 4, HADF processes an LR field $\boldsymbol{X} \in \mathbb{R}^{C\times D_{\mathrm{lr}}\times H_{\mathrm{lr}}\times W_{\mathrm{lr}}}$ and its corresponding LR spatial coordinates $P_{\mathrm{lr}} = (x_{\mathrm{lr}}, y_{\mathrm{lr}}, z_{\mathrm{lr}}, \frac{1}{\mathrm{scale\_factor}})$ as inputs. Here, C denotes the number of physical channels (e.g., density and velocity components), and $(D_{\mathrm{lr}}, H_{\mathrm{lr}}, W_{\mathrm{lr}})$ represent the spatial dimensions along the three coordinates. The coordinates $(x_{\mathrm{lr}}, y_{\mathrm{lr}}, z_{\mathrm{lr}}) \in [-1, 1]^3$ define the spatial positions of LR grid points, while scale_factor specifies the super-resolution ratio between the LR and HR fields. Given these inputs, HADF first extracts field content features and spatial features using a feature encoding module composed of a residual dense network (RDN) and a hash-adaptive spatial encoding mechanism. These features are subsequently fused through a dynamic feature fusion module, enabling the network model to capture intricate turbulent structures via cross-domain interactions. The fused features are then processed by an implicit decoding module, which predicts an HR output field $\boldsymbol{Y} \in \mathbb{R}^{C\times D_{\mathrm{hr}}\times H_{\mathrm{hr}}\times W_{\mathrm{hr}}}$ corresponding to the desired spatial resolution $D_{\mathrm{hr}}, H_{\mathrm{hr}}, W_{\mathrm{hr}}$. This prediction is achieved by querying arbitrary HR coordinates $\boldsymbol{P}_{\mathrm{hr}} = (x_{\mathrm{hr}},\ y_{\mathrm{hr}},\ z_{\mathrm{hr}}) \in [-1, 1]^3$, represented by the red sphere in Fig. 5. The entire process can be represented by the following formula:

$$\boldsymbol{Y} = I\Big(F\big(H(\boldsymbol{P}_{\mathrm{lr}}), R(\boldsymbol{X})\big), \boldsymbol{P}_{\mathrm{hr}}\Big), \tag{1}$$

where R, H, F, and I denote the RDN feature encoder, hash adaptive spatial encoder, dynamic feature fusion module, and implicit decoder, respectively. Each voxel in the turbulent field forms a spatial volume, with its center, represented by blue spheres in Fig. 5, serving as the reference coordinates, improving spatial alignment during reconstruction. To ensure computational stability, we add zero padding along the spatial boundaries to maintain the output size during convolution.

Additionally, as illustrated on the right side of Fig. 4, LR consistency loss is introduced to support robust training under partially unavailable HR data. It uses the consistency between downsampled predictions and the original LR inputs to guide the network model under incomplete supervision.

3.1 Feature encoding

To effectively encode turbulent flow characteristics, our HADF network model incorporates two specialized modules for feature extraction. The 3D RDN extracts field content features from LR turbulence data. Concurrently, the hash-adaptive spatial encoding module extracts spatial features, emphasizing critical turbulent regions to effectively capture non-uniform flow characteristics (Park et al., 2021a, 2021b).

As depicted in Fig. 4, hash-adaptive spatial encoding transforms the LR space $\boldsymbol{P}_{\mathrm{lr}} =$

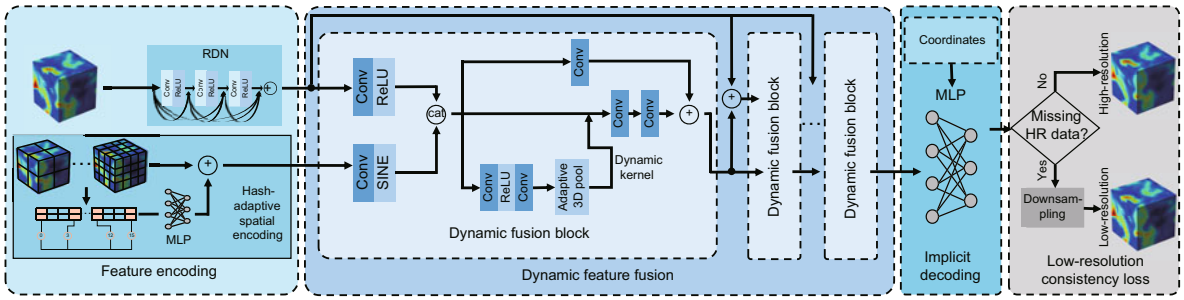

Fig. 4 Architecture of HADF

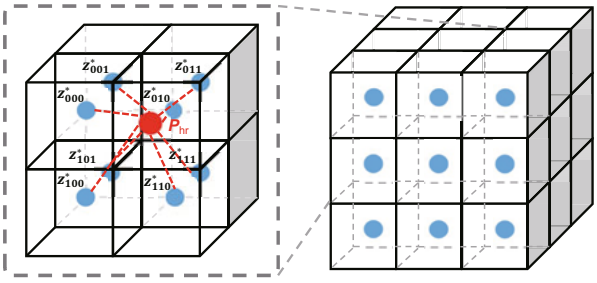

Fig. 5 HADF representation with local ensemble. A continuous turbulent field is represented as a 3D feature map with implicit decoding. The HR turbulent field is predicted by an ensemble of local predictions, which preserves turbulent details across different regions. References to color refer to the online version of this figure

$(x_{\mathrm{lr}}, y_{\mathrm{lr}}, z_{\mathrm{lr}}, \frac{1}{\mathrm{scale\_factor}})$ into a multi-resolution feature grid comprising $L = 5$ hierarchical levels. Each grid vertex stores learnable features of dimension $F = 2$, with the resolution of the l^{th} layer N_l defined by

$$N_l = \lfloor N_{\min} \cdot b^l \rfloor, \quad b = \exp\left(\frac{\ln N_{\max} - \ln N_{\min}}{L-1}\right), \tag{2}$$

where $N_{\min} = 16$, $N_{\max} = 512$, and $l = 0, 1, \ldots, L-1$. For input coordinates $\boldsymbol{P}_{\mathrm{lr}}$ at the l^{th} layer of the multi-resolution feature grid, the coordinates are scaled by the layer's resolution. The spatial features are then obtained via trilinear interpolation from the neighboring grid vertices, as illustrated in Fig. 4. To identify these neighboring vertices, the coordinates $\boldsymbol{P}_{\mathrm{lr}}$ are rounded down and up as follows:

$$\lfloor \boldsymbol{P}^l_{\mathrm{lr}} \rfloor = \lfloor \boldsymbol{P}_{\mathrm{lr}} \cdot N_l \rfloor, \quad \lceil \boldsymbol{P}^l_{\mathrm{lr}} \rceil = \lceil \boldsymbol{P}_{\mathrm{lr}} \cdot N_l \rceil. \tag{3}$$

Each vertex is mapped to an entry in the layer's feature vector array, which has a fixed capacity of at most T. At coarser levels, where the number of grid vertices is smaller than T, the mapping is direct, allowing features to be retrieved by their indices. At finer resolutions, where the vertex count exceeds T, a hash function is employed:

$$h(P^l_{\mathrm{lr}}) = \left(\bigoplus_{i=1}^{4} x_i \pi_i\right) \bmod T, \tag{4}$$

where $\bigoplus$ denotes the bitwise exclusive or (XOR) operation, and $\{\pi_i\}$ are distinct large prime numbers (Müller et al., 2022).

The operations described above are represented as a four-dimensional (4D) hash encoding function $\gamma_{4\mathrm{D}}(\boldsymbol{P}_{\mathrm{lr}})$, which maps arbitrary coordinates $\boldsymbol{P}_{\mathrm{lr}}$ to high-dimensional spatial features. These features are subsequently mapped to a 4D deformation vector $\Delta\boldsymbol{P}_{\mathrm{lr}}$ through a lightweight multi-layer perceptron (MLP): $\gamma_{4\mathrm{D}}(\boldsymbol{P}_{\mathrm{lr}}) \to \Delta\boldsymbol{P}_{\mathrm{lr}}$. The final spatial features of the hash-adaptive spatial encoding module are computed as

$$H(\boldsymbol{P}_{\mathrm{lr}}) = \boldsymbol{P}_{\mathrm{lr}} + \Delta\boldsymbol{P}_{\mathrm{lr}}. \tag{5}$$

This hash-adaptive spatial encoding module is adaptively adjusted to turbulent flow fields, accommodating complex turbulent variations and enhancing focus on critical turbulent regions.

3.2 Dynamic feature fusion

As illustrated in Fig. 4, our dynamic feature fusion module adaptively integrates content features $R(\boldsymbol{X})$ with spatial features $H(\boldsymbol{P}_{\mathrm{lr}})$ to effectively capture both local and global characteristics of turbulent flow fields. Initially, the input features are processed through nonlinear transformations to produce fused representations $\boldsymbol{R}_0$ and $\boldsymbol{H}_0$:

$$\boldsymbol{R}_0 = \mathrm{ReLU}(\mathrm{Conv}(R(\boldsymbol{X}))), \tag{6}$$

$$\boldsymbol{H}_0 = \mathrm{SineAct}(\mathrm{Conv}(H(\boldsymbol{P}_{\mathrm{lr}}))). \tag{7}$$

At each fusion level i, the features $\boldsymbol{R}_i$ and $\boldsymbol{H}_i$ are concatenated and processed to generate dynamic convolution kernel parameters:

$$\boldsymbol{\Phi}_i = \text{Adaptive3dPool}(\text{Conv}(\text{ReLU}(\text{Conv}([\boldsymbol{R}_i \oplus \boldsymbol{H}_i])))). \quad (8)$$

Here, $\boldsymbol{\Phi}_i$ represents the kernel parameters after global pooling. These parameters are then reshaped into 3D convolution kernels:

$$\boldsymbol{K}_i = \text{Reshape}(\boldsymbol{\Phi}_i) \in \mathbb{R}^{C\times 3\times 3\times 3}, \quad (9)$$

where C is the channel dimension of the concatenated features. Because $\boldsymbol{K}_i$ is generated directly from the fused features $[\boldsymbol{R}_i \oplus \boldsymbol{H}_i]$, the resulting convolution kernels are not fixed but vary adaptively with the feature distribution at each fusion level. These adaptive kernels $\boldsymbol{K}_i$ are tailored to the flow field's unique characteristics, enabling enhanced integration of content and spatial features for superior feature representation compared to standard convolutions. The kernels drive group convolutions on the concatenated features $[\boldsymbol{R}_i \oplus \boldsymbol{H}_i]$, with a residual connection processing the same features and adding the result to the convolution output, thus stabilizing training and mitigating gradient issues, to produce updated spatial features:

$$\boldsymbol{H}_{i+1} = \text{Conv}(\text{Conv}([\boldsymbol{R}_i \oplus \boldsymbol{H}_i], \boldsymbol{K}_i)) + \text{Conv}([\boldsymbol{R}_i \oplus \boldsymbol{H}_i]). \quad (10)$$

The fusion process refines content features by combining the updated spatial features with the original content features to produce $\boldsymbol{R}_{i+1}$:

$$\boldsymbol{R}_{i+1} = \text{ReLU}(\text{Conv}([\boldsymbol{H}_{i+1} \oplus R(\boldsymbol{X})])). \quad (11)$$

The final output of the dynamic feature fusion module is

$$F(H(\boldsymbol{P}_{\text{lr}}), R(\boldsymbol{X})) = \boldsymbol{H}_{i+1}. \quad (12)$$

3.3 Implicit decoding

In this subsection, we employ an implicit decoding module that maps coordinates $\boldsymbol{P}_{\text{hr}} = (x_{\text{hr}},\ y_{\text{hr}},\ z_{\text{hr}})$ and corresponding fused features to high-fidelity turbulent values, enabling arbitrary resolution reconstruction of turbulent flow fields. We integrate a 3D local ensemble strategy and resolution-aware cell decoding approach to enhance the final reconstruction quality. The decoder processes coordinates $\boldsymbol{P}_{\text{hr}}$ and fused features $F(H(\boldsymbol{P}_{\text{lr}}), R(\boldsymbol{X}))$, mapping them to a continuous space through

$$I\Big(F\big(H(\boldsymbol{P}_{\text{lr}}), R(\boldsymbol{X})\big), \boldsymbol{P}_{\text{hr}}\Big) = \text{MLP}\left(\boldsymbol{z}^*,\ \boldsymbol{P}_{\text{hr}} - \boldsymbol{v}^*\right), \quad (13)$$

where $\boldsymbol{z}^*$ represents the fused feature closest to $\boldsymbol{P}_{\text{hr}}$, and $\boldsymbol{v}^*$ is its spatial position. Take Fig. 5 as an example; $\boldsymbol{z}^*_{010}$ is the $\boldsymbol{z}^*$ for $\boldsymbol{P}_{\text{hr}}$ in our current definition, whereas $\boldsymbol{v}^*$ is defined as the coordinate for $\boldsymbol{z}^*_{010}$.

The 3D local ensemble strategy (Chen YB et al., 2021) reduces artifacts in nearest neighbor sampling. As depicted in Fig. 5, we extend the decoding process (13) by aggregating the eight surrounding voxel vertices at each query point $\boldsymbol{P}_{\text{hr}}$. This aggregation employs a weighted sum of HR turbulent values, formalized as

$$\begin{aligned} &I\Big(F\big(H(\boldsymbol{P}_{\text{lr}}), R(\boldsymbol{X})\big), \boldsymbol{P}_{\text{hr}}\Big) \\ &= \sum_{t\in T} \frac{V_n}{V} \cdot \text{MLP}\left(\boldsymbol{z}^*_n,\ \boldsymbol{P}_{\text{hr}} - \boldsymbol{v}^*_n\right), \end{aligned} \quad (14)$$

where $\boldsymbol{z}^*_n$ ($n \in N = \{000, 001, \ldots, 111\}$) denotes the latent feature surrounding the query point $\boldsymbol{P}_{\text{hr}}$, and $\boldsymbol{v}^*_n$ are its spatial coordinates. The volumetric weight V_t is determined by the volume of the region between $\boldsymbol{v}^*_n$ and $\boldsymbol{P}_{\text{hr}}$, calculated using the coordinate differences ($\Delta x,\ \Delta y,\ \Delta z$) along each axis. These weights are normalized by $V = \sum_{n\in N} V_n$, ensuring balanced contributions across vertices. To manage query points near the domain boundaries, we apply mirror padding to the feature map, ensuring that the aggregation process functions effectively across the entire spatial domain.

Resolution-aware cell decoding enhances spatial consistency across scales. Unlike conventional methods treating query points as zero-volume entities, we incorporate a cell size vector as an auxiliary input to model voxel structures across resolutions, improving cross-scale robustness. Eq. (13) is reformulated as

$$I\Big(F\big(H(\boldsymbol{P}_{\text{lr}}), R(\boldsymbol{X})\big), \boldsymbol{P}_{\text{hr}}\Big) = \text{MLP}\left(\boldsymbol{z}^*_n,\ \boldsymbol{P}_{\text{hr}} - \boldsymbol{v}^*_n,\ \boldsymbol{c}\right), \quad (15)$$

where $\boldsymbol{c} = [c_{\text{d}}, c_{\text{h}}, c_{\text{w}}]$ contains three values that specify the depth, height, and width of the query pixel separately.

3.4 LR consistency loss

Super-resolution models for turbulent flow reconstruction typically require paired LR and HR data, which poses challenges when HR data are partially unavailable. To address this, we propose an LR consistency loss that facilitates training with unpaired data, ensuring the reconstruction of fine-scale turbulent structures.

Given an LR input field $\boldsymbol{X} \in \mathbb{R}^{C\times D_{\mathrm{lr}}\times H_{\mathrm{lr}}\times W_{\mathrm{lr}}}$, the network model predicts an HR field $\hat{\boldsymbol{Y}} \in \mathbb{R}^{C\times D_{\mathrm{hr}}\times H_{\mathrm{hr}}\times W_{\mathrm{hr}}}$. A downsampling operator D is applied to $\hat{\boldsymbol{Y}}$, yielding an LR counterpart $\hat{\boldsymbol{X}} \in \mathbb{R}^{C\times D_{\mathrm{lr}}\times H_{\mathrm{lr}}\times W_{\mathrm{lr}}}$. The consistency loss is defined as

$$L_{\mathrm{lr}} = \|\hat{\boldsymbol{X}} - \boldsymbol{X}\|, \tag{16}$$

where $\|\cdot\|$ denotes an L_2 norm. This loss ensures that the downsampled prediction aligns with the input $\boldsymbol{X}$, enforcing physically consistent reconstructions.

4 Experiments

In this section, we present our evaluation of the performance of HADF in turbulent super-resolution reconstruction. Section 4.1 presents the experimental datasets. Section 4.2 defines the evaluation metrics. Section 4.3 reports the experimental results.

To ensure fair and reproducible comparisons, all baseline models are implemented following consistent principles. Feature extraction in local implicit image function (Liif) (Chen YB et al., 2021), Liif-sine (Sitzmann et al., 2020), dual interactive implicit neural network (Diinn) (Nguyen and Beksi, 2023), and the proposed HADF is performed using a 3D residual RDN structure. After this stage, each method applies its own decoder. Liif and Liif-sine employ implicit neural functions, Diinn adopts a diffusion-based architecture, and HADF incorporates the adaptive decoder developed in this study. The hypernetwork (Hyper) (Zhang et al., 2023) model does not include an RDN-based encoder, and the implementation follows the description provided in the original paper. The overall model configuration is adjusted so that the parameter scale remains comparable to those of the other baseline methods, which ensures fairness in the evaluation process.

The experiments are conducted using the Adam optimizer with an initial learning rate of 1×10^{-4}. A StepLR scheduler reduces the learning rate by a factor of 0.5 every 300 steps. Training is performed for 500 epochs on NVIDIA A100 GPUs with 40 GB memory.

4.1 Experimental data description

To evaluate the performance of HADF, we adopt three direct numerical simulation (DNS) datasets from the BLASTNet database (Chung et al., 2023a, 2023b). All datasets are organized into 3D volumes with a resolution of $128 \times 128 \times 128$, where each grid point contains four channels corresponding to density and the three velocity components (u, v, w). Both the interpolation and extrapolation experiments are based on the BLASTNet Momentum128 3D SR dataset (Chung et al., 2022), which contains volumetric data cropped from DNS of turbulent flows. This dataset is a curated subset extracted from HR DNS cases in the original BLASTNet database and therefore does not possess explicit temporal sequence information.

The interpolation task employs the slot-burner hydrogen jet flame dataset. The configuration corresponds to a turbulent lifted jet flame with Reynolds number $Re_{\mathrm{jet}} = 8000$, where velocity fluctuations are approximately 10% of the bulk jet velocity (Jung et al., 2021). The flow field represents mixing between a high-speed hydrogen–nitrogen jet and heated coflow air. A total of 82 samples are used, with 65 allocated to training, 8 to validation, and 9 to testing.

In the case of extrapolation, the hydrogen–air premixed turbulent flame dataset is used. This dataset is characterized by a fixed Karlovitz number $Ka_{\mathrm{u}} = 6.8$ and inflow velocity ratio $U_{\mathrm{in}}/S_L = 2.45, 3.67, 4.63, 5.51$ (Savard et al., 2019). It represents freely propagating turbulent flames under autoignitive reheat combustion conditions and captures turbulence–chemistry interactions as well as transitions between spontaneous ignition and deflagration regimes. A total of 181 samples are used, with 145 for training, 13 for validation, and 23 for testing.

For the reconstruction task under conditions of partial HR data unavailability and noisy perturbations, we adopt the methane–oxygen inert homogeneous isotropic turbulence (HIT) dataset (Chung et al., 2022). The original DNS resolution is $129 \times 129 \times 129$, and the outermost layer is removed to obtain $128 \times 128 \times 128$ volumes for consistency. This dataset represents a canonical isotropic turbulence case without combustion. A total of 99 spatial samples are employed, with 79 used for training, 10 for validation, and 10 for testing. Unlike the interpolation and extrapolation datasets, this dataset provides temporal evolution in the form of 98 consecutive snapshots.

4.2 Evaluation metrics

To quantitatively assess the performance of HADF, we employ peak signal-to-noise ratio (PSNR) and physical metrics to evaluate the fidelity and physical consistency of reconstructed flow fields.

PSNR is a standard image processing metric, defined as

$$\mathrm{PSNR} = 10\lg\left(\frac{L^2}{\mathrm{MSE}}\right), \tag{17}$$

where L represents the maximum value of data such as density and velocity, and MSE denotes the mean-squared error between predicted and reference values. A higher PSNR value indicates a reduced error in the reconstructed field.

We define Metrics = {SSIM, NRMSE} to quantify field characteristics. The structural similarity index (SSIM) measures statistical similarity using mean, variance, and covariance. SSIM equals 1 for identical fields and 0 for dissimilar fields. The normalized root-mean-squared error (NRMSE) quantifies pointwise deviations, normalized by the reference data's dynamic range. For multi-channel physical quantities, specifically density ρ and velocity components $\boldsymbol{u}$, we define composite metrics as

$$\mathrm{Metric}_{\rho,\boldsymbol{u}} \equiv \frac{1}{4}\left[\mathrm{Metric}(\hat{\rho},\rho) + \sum_{i=1}^{3}\mathrm{Metric}(\hat{u}_i,u_i)\right], \tag{18}$$

$$\mathrm{Metric}_{\mathrm{sgs}} \equiv \frac{1}{3}\sum_{k=1}^{3}\mathrm{Metric}\left((\nabla\cdot\hat{\boldsymbol{\tau}}^{\mathrm{sgs}})_k,(\nabla\cdot\boldsymbol{\tau}^{\mathrm{sgs}})_k\right), \tag{19}$$

where $\hat{\cdot}$ denotes predicted quantities. The metric $\mathrm{Metric}_{\rho,\boldsymbol{u}}$ averages performance across each channel. The metric $\mathrm{Metric}_{\mathrm{sgs}}$ evaluates super-resolution suitability for turbulence modeling in coarse grid simulations by measuring the accuracy of the subgrid-scale (SGS) stress divergence $\nabla\cdot\boldsymbol{\tau}^{\mathrm{sgs}}$, which captures physical information lost during coarse graining.

4.3 Performance evaluation

We train all models on super-resolution scaling factors from $\times2$ to $\times4$ and conduct multi-scale evaluations across interpolation, extrapolation, and partial HR data unavailable scenarios, covering trained scales $\times2$ to $\times4$ and untrained scales $\times8$ to $\times32$. A recent study (Fukami et al., 2024) has shown that convolutional network-based models can effectively learn the scale-invariant properties of turbulence, and we demonstrate our model's scale invariance by evaluating it with LR inputs of varying resolutions. Results are compared with those of baselines Trilinear interpolation, Liif, Liif-sine, Diinn, and Hyper. Trilinear interpolation consistently exhibits lower PSNR than other methods despite high physical metrics, as evidenced in Tables 1–4, precluding its inclusion in physical metric comparisons across all tables. Furthermore, error visualizations provide clear evidence of its substantial reconstruction error. These visualizations are generated by extracting a representative 2D slice from the 3D turbulent flow datasets to better illustrate the details of the reconstruction (Fig. 6).

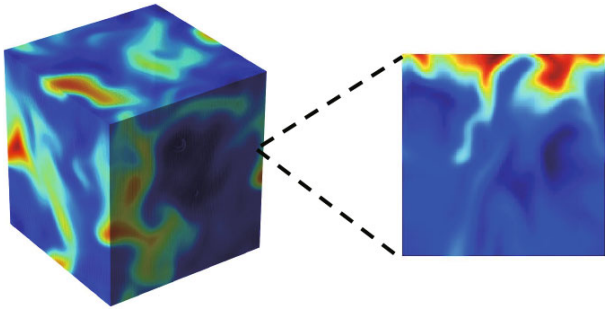

Fig. 6 Two-dimensional slice extracted from three-dimensional turbulent flow datasets

4.3.1 Interpolation experiment

The interpolation experiment evaluates HADF's performance using slot-burner hydrogen jet flame data, where training and testing data share the same operational regime to assess interpolation accuracy within established physical mechanisms. The dataset is partitioned into 80% for training, 10% for validation, and 10% for testing to ensure comprehensive performance evaluation.

HADF demonstrates superior super-resolution performance in interpolation for turbulent flow reconstruction, outperforming baseline methods across most metrics at all scaling factors (Tables 1 and 2). Although HADF exhibits slightly lower $\mathrm{SSIM}_{\mathrm{sgs}}$ at $\times2$ and $\mathrm{NRMSE}_{\mathrm{sgs}}$ at $\times32$ compared to Liif-sine, it achieves a significant PSNR advantage, balancing local physical accuracy and global fidelity. Fig. 7 compares density field reconstructions from all models against HR GT using LR at scaling factors $\times2$, $\times4$, $\times6$, and $\times8$, with corresponding error distributions

Table 1 Performance comparison of interpolation testing at trained scales (×2 to ×4)

Method	×2					×3				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	16.0781	0.9885	0.8142	0.0031	0.4533	4.2561	0.9286	0.2973	0.0414	0.7584
Liif (Chen YB et al., 2021)	34.5702	0.8207	0.1995	0.0030	0.4527	30.1593	0.7799	0.1452	0.0144	0.6769
Liif-sine (Sitzmann et al., 2020)	21.1208	0.7222	**0.3395**	0.0278	0.3740	21.0640	0.6989	0.1948	0.0370	0.6236
Diinn (Nguyen and Beksi, 2023)	34.3560	0.8902	0.0026	0.0069	1.0000	29.9198	0.8387	0.0988	0.0259	1.2930
Hyper (Zhang et al., 2023)	12.6918	0.4626	0.0064	0.1674	1.0000	12.4206	0.4469	0.0024	0.1777	1.6644
HADF	**38.8819**	**0.9116**	0.2703	**0.0008**	**0.2269**	**33.4629**	**0.8665**	**0.2140**	**0.0096**	**0.5635**

Method	×4				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	7.5645	0.9497	0.4729	0.0219	0.7282
Liif (Chen YB et al., 2021)	30.8135	0.7614	0.1381	0.0185	0.7400
Liif-sine (Sitzmann et al., 2020)	21.4157	0.6947	0.2314	0.0359	0.6215
Diinn (Nguyen and Beksi, 2023)	29.5645	0.8149	0.0090	0.0293	1.0001
Hyper (Zhang et al., 2023)	12.6311	0.4358	0.0046	0.1841	1.0000
HADF	**35.3708**	**0.8670**	**0.2524**	**0.0070**	**0.5121**

Bold values indicate the best results. Trilinear interpolation is excluded from physical metrics comparison due to its consistently much lower PSNR than other methods. HADF: hash-adaptive dual fusion; NRMSE: normalized root-mean-squared error; PSNR: peak signal-to-noise ratio (in dB); SSIM: structural similarity index

Table 2 Performance comparison of interpolation testing at untrained scales (×8 to ×32)

Method	×8					×16				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	1.9262	0.8725	0.4200	0.0829	0.8717	–0.9635	0.7919	0.1813	0.1969	0.9307
Liif (Chen YB et al., 2021)	26.5966	0.6661	0.0953	0.0827	0.8940	21.7465	0.5765	0.0346	0.2181	0.9477
Liif-sine (Sitzmann et al., 2020)	20.9740	0.6324	0.1808	0.0772	0.7914	20.6002	0.5475	0.1097	0.1781	0.8645
Diinn (Nguyen and Beksi, 2023)	25.0628	0.6965	0.0010	0.0918	1.0001	21.4954	0.6045	0.00003	0.2073	1.0000
Hyper (Zhang et al., 2023)	12.5341	0.3891	0.0038	0.2283	1.0936	12.5042	0.3487	0.0083	0.3108	1.0114
HADF	**31.7067**	**0.7850**	**0.2253**	**0.0449**	**0.7219**	**26.3487**	**0.6753**	**0.1821**	**0.1498**	**0.8309**

Method	×32				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	–0.0721	0.7075	0.0465	0.3354	0.9598
Liif (Chen YB et al., 2021)	17.1611	0.4903	0.0443	0.3794	0.9840
Liif-sine (Sitzmann et al., 2020)	19.9894	0.4693	0.0905	0.3347	**0.9639**
Diinn (Nguyen and Beksi, 2023)	18.7937	0.5117	0.0001	0.3531	1.0003
Hyper (Zhang et al., 2023)	12.5505	0.3317	0.0777	0.4211	0.9970
HADF	**20.9539**	**0.5526**	**0.1113**	**0.3184**	0.9662

Bold values indicate the best results. Trilinear interpolation is excluded from physical metrics comparison due to its consistently much lower PSNR than other methods. HADF: hash-adaptive dual fusion; NRMSE: normalized root-mean-squared error; PSNR: peak signal-to-noise ratio (in dB); SSIM: structural similarity index

Table 3 Performance comparison of extrapolation testing at trained scales (×2 to ×4)

Method	×2					×3				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	11.5397	0.9793	0.8264	0.0012	0.2109	–1.8810	0.8861	0.4717	0.0242	0.5611
Liif (Chen YB et al., 2021)	38.8482	0.8777	0.6006	0.0024	0.1232	34.3896	0.8503	0.4637	0.0064	0.3627
Liif-sine (Sitzmann et al., 2020)	29.1481	0.8203	0.6284	0.0136	0.1239	28.2080	0.8063	0.4541	0.0173	0.3716
Diinn (Nguyen and Beksi, 2023)	35.2865	0.8480	0.0280	0.0139	9.2812	30.5785	0.7764	0.1219	0.0242	3.5912
Hyper (Zhang et al., 2023)	34.2279	0.8892	0.0103	0.0040	1.0000	31.6141	0.8425	0.0039	0.0087	1.2359
HADF	**42.3701**	**0.9343**	**0.7007**	**0.0005**	**0.0730**	**35.4116**	**0.9002**	**0.4974**	**0.0045**	**0.3604**

Method	×4				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	2.2732	0.9348	0.5675	0.0104	0.4806
Liif (Chen YB et al., 2021)	37.0581	0.8601	0.5678	0.0035	0.2589
Liif-sine (Sitzmann et al., 2020)	29.0184	0.8085	0.5728	0.0139	0.2484
Diinn (Nguyen and Beksi, 2023)	30.4215	0.7585	0.0242	0.0253	2.6844
Hyper (Zhang et al., 2023)	29.7633	0.7855	0.0033	0.0150	1.0509
HADF	**41.0476**	**0.9245**	**0.6234**	**0.0010**	**0.2092**

Bold values indicate the best results. Trilinear interpolation is excluded from physical metrics comparison due to its consistently much lower PSNR than other methods. HADF: hash-adaptive dual fusion; NRMSE: normalized root-mean-squared error; PSNR: peak signal-to-noise ratio (in dB); SSIM: structural similarity index

Table 4 Performance comparison of extrapolation testing at untrained scales (×8 to ×32)

Method	×8					×16				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	–4.4623	0.7949	0.3247	0.0528	0.7402	–9.4860	0.6111	0.1348	0.1756	0.9070
Liif (Chen YB et al., 2021)	31.3179	0.7723	0.4074	0.0182	0.5186	24.7493	0.5720	0.1434	0.1169	0.7781
Liif-sine (Sitzmann et al., 2020)	27.2669	0.7461	0.4441	0.0252	0.4702	22.9585	0.5741	0.1783	0.1059	0.7277
Diinn (Nguyen and Beksi, 2023)	25.7878	0.5883	0.0262	0.0645	1.4615	21.7765	0.4446	0.0156	0.1719	1.2599
Hyper (Zhang et al., 2023)	25.3257	0.5918	0.0025	0.0522	1.0019	21.7042	0.4443	0.0016	0.1505	1.0000
HADF	**34.2766**	**0.8510**	**0.5027**	**0.0096**	**0.4261**	**26.1667**	**0.6494**	**0.2196**	**0.0780**	**0.6871**

Method	×32				
	PSNR	$SSIM_{\rho,\boldsymbol{u}}$	$SSIM_{sgs}$	$NRMSE_{\rho,\boldsymbol{u}}$	$NRMSE_{sgs}$
Trilinear	–12.5248	0.4666	0.0709	0.3564	0.9823
Liif (Chen YB et al., 2021)	19.9655	0.3935	0.0202	0.3446	0.9542
Liif-sine (Sitzmann et al., 2020)	19.1087	0.4040	0.0500	0.3174	0.9248
Diinn (Nguyen and Beksi, 2023)	18.5775	0.3676	0.0343	0.3513	1.1111
Hyper (Zhang et al., 2023)	18.8732	0.3800	0.0026	0.3166	0.9994
HADF	**20.7102**	**0.4401**	**0.0721**	**0.2700**	**0.9142**

Bold values indicate the best results. Trilinear interpolation is excluded from physical metrics comparison due to its consistently much lower PSNR than other methods. HADF: hash-adaptive dual fusion; NRMSE: normalized root-mean-squared error; PSNR: peak signal-to-noise ratio (in dB); SSIM: structural similarity index

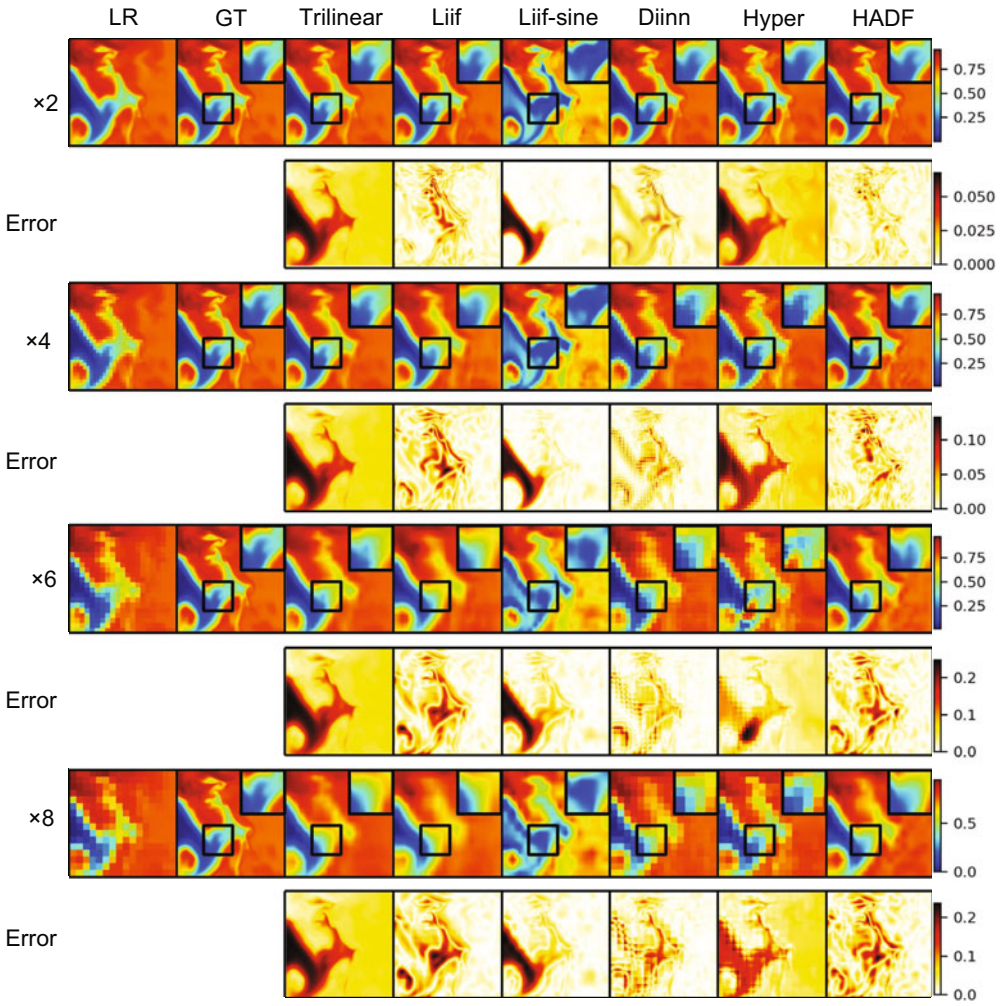

Fig. 7 Qualitative comparison of interpolation testing

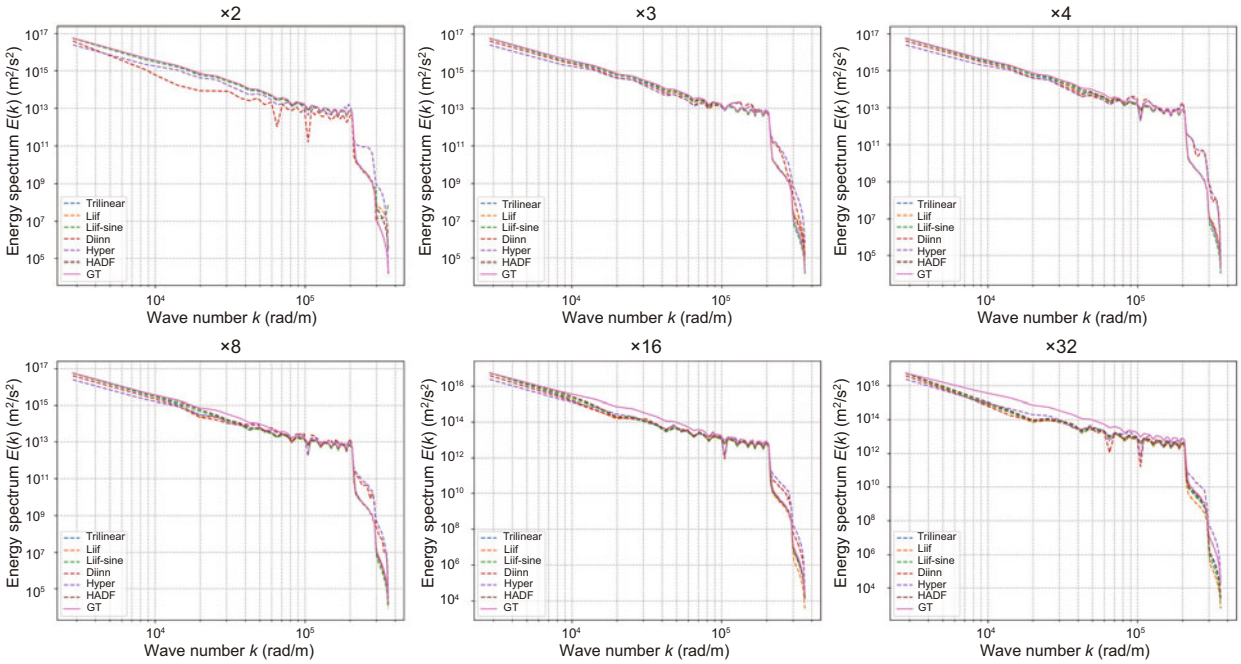

Fig. 8 Kinetic energy spectrum comparison for interpolation testing (References to color refer to the online version of this figure)

(Error). Trilinear interpolation relies on numerical operations over adjacent LR data points, neglecting the nonlinear complexity of turbulent flows. This results in significant reconstruction errors, observable in error distributions across all scaling factors. Diinn and Hyper perform initial interpolation to upsample LR data before reconstruction, introducing discontinuities and granularity in the output. These discontinuities become more apparent at higher scaling factors, despite occasionally with lower error metrics. Liif and Liif-sine improve detail recovery but only extract content features from LR data. They overlook complex spatial features critical for turbulent structures, resulting in reduced detail fidelity compared to HADF.

In addition, we assess the reconstructions from a spectral perspective using the kinetic energy spectrum (Fig. 8). The spectrum quantifies how turbulent kinetic energy is distributed across spatial scales. The horizontal axis is the wave number k (in rad/m), inversely proportional to spatial scale, and the vertical axis is the energy spectrum $E(k)$ (in $\mathrm{m}^2/\mathrm{s}^2$). As the scaling factor increases from $\times 2$ to $\times 32$, Trilinear remains consistently below the GT from low to high k, with deviations widening, which reflects typical interpolation-induced information loss. Liif and Liif-sine are lower than GT at low and intermediate k, and at high k their spectra fluctuate around GT, exhibiting incorrect deviations in the small-scale range. Diinn follows the decay trend of GT at low k but with biased magnitude. Its spectrum shows strong oscillations from intermediate to high k and exceeds GT significantly at high k, indicating insufficient capability in accurately reconstructing small-scale structures. Hyper shows a similar pattern to Diinn, with an even greater overshoot in the high-k region. In contrast, HADF achieves the closest agreement with GT across the full spectral range, with a smoother decay and without oscillations or premature attenuation, demonstrating clear advantages in turbulent flow reconstruction.

In contrast, HADF employs a hash-adaptive spatial encoding strategy and dynamic feature fusion to prioritize critical turbulent regions and integrate content and spatial features effectively. These mechanisms enable HADF to reconstruct HR turbulent fields with the minimum errors, excellent spatial continuity, and superior detail fidelity. Furthermore, HADF achieves high visual accuracy and maintains close agreement with the GT in the kinetic energy spectrum. This consistency across both spatial and spectral domains demonstrates its clear effectiveness in resolving fine-scale turbulent structures across diverse scaling factors.

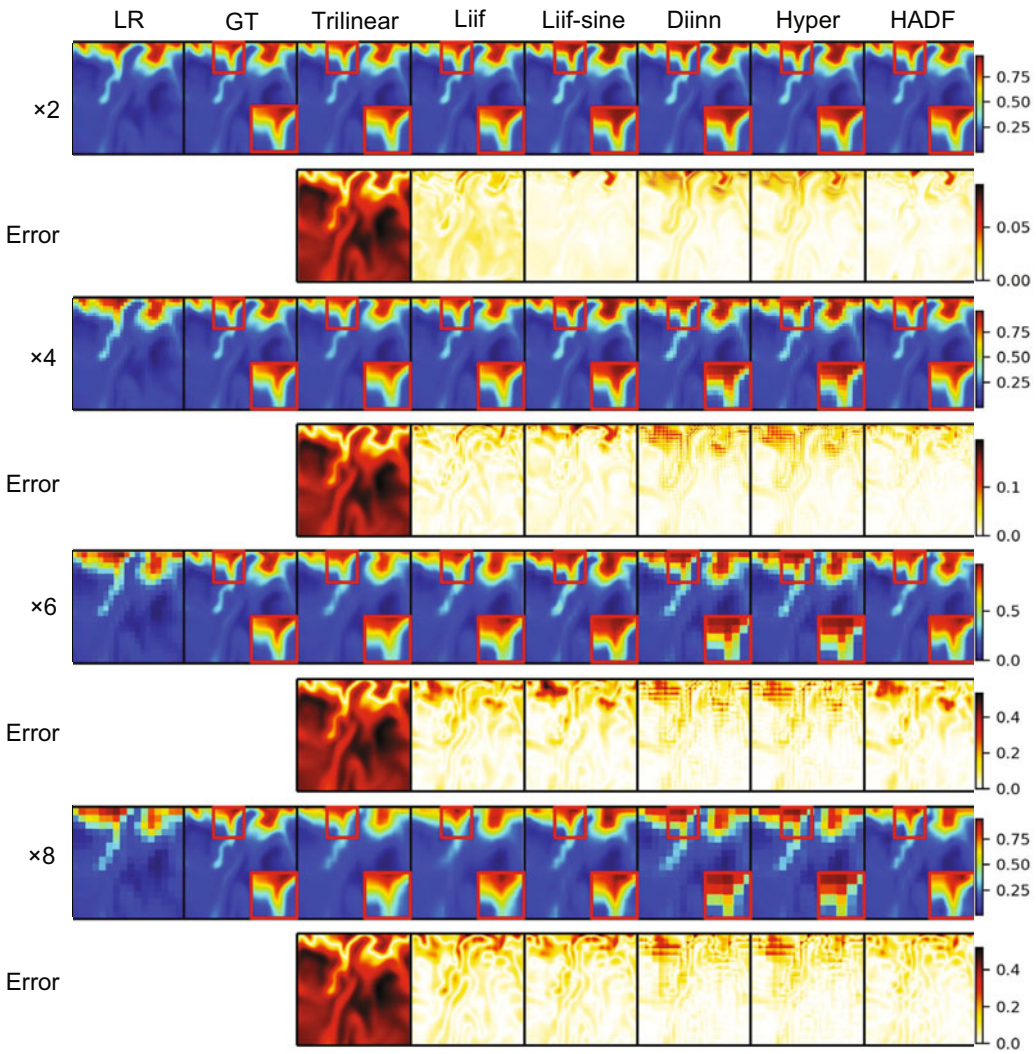

Fig. 9 Qualitative comparison of extrapolation testing

4.3.2 Extrapolation experiment

The extrapolation experiment evaluates HADF's generalization using hydrogen–air premixed freely propagating turbulent flame data. Training and validation data encompass cases with a fixed Karlovitz number ($Ka_{\mathrm{u}} = 6.8$) at inflow velocity ratio of $U_{\mathrm{in}}/S_L = 3.67$, 4.63, and 5.51, with the dataset partitioned into 90% for training and 10% for validation. The test data consist of an extreme condition at $U_{\mathrm{in}}/S_L = 2.45$ to assess performance beyond the training parametric space.

As shown in Tables 3 and 4, HADF achieves the highest performance in extrapolation, outperforming baselines across all metrics, demonstrating robust generalization to unseen conditions. Fig. 9 illustrates that, similar to its performance in interpolation tasks, Trilinear interpolation fails to recover HR turbulent structures in extrapolation tasks. Diinn and Hyper still produce coarse reconstructions with evident detail loss. Although Liif and Liif-sine demonstrate some reconstruction capability, they remain limited in preserving details. In contrast, HADF exhibits excellent generalization, accurately capturing fine-scale structures and overall distributions across all scales, validating its superior performance in handling unknown conditions.

Fig. 10 further shows that, unlike in interpolation tasks, none of the methods can accurately recover the high-wavenumber range of the energy spectrum, with all curves deviating from the GT beyond 8×10^4. This phenomenon merits further exploration

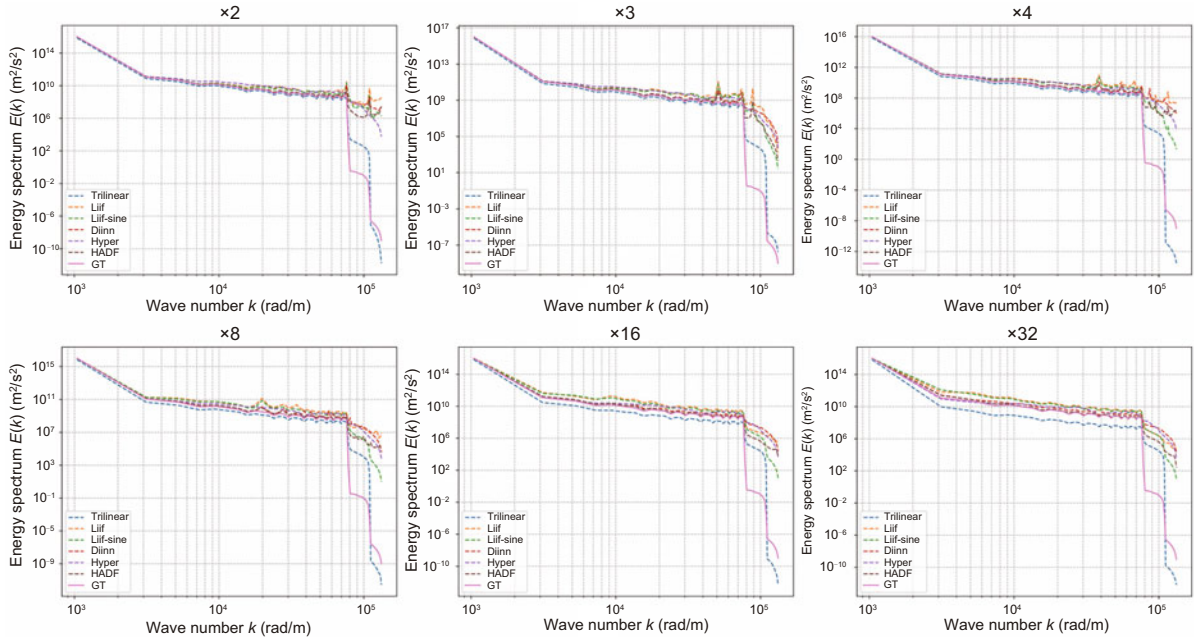

Fig. 10 Kinetic energy spectrum comparison for extrapolation testing (References to color refer to the online version of this figure)

in future studies. Moreover, Trilinear interpolation fails to recover HR turbulent structures in extrapolation scenarios, with its spectrum consistently below GT in the low-to-intermediate wavenumber range. Liif, Liif-sine, Diinn, and Hyper still produce coarse reconstructions, showing clear detail loss in the low-to-intermediate k range and exhibiting spectra significantly higher than GT. In contrast, although HADF shows some spectral error beyond 8×10^4, it demonstrates excellent generalization in the low-to-intermediate wavenumber range, accurately capturing detail structures and overall energy distributions, with its spectrum curve closely matching GT.

4.3.3 Unpaired data experiments

To evaluate the advantages of the LR consistency loss in scenarios with unpaired data, we conduct two sets of experiments to analyze the performance of HADF in turbulent flow super-resolution reconstruction.

In the first experiment, we train HADF with LR consistency loss and compare it against baseline methods Trilinear, Liif, Liif-sine, Diinn, and Hyper under a 30% HR data unavailability rate, across scaling factors ×2, ×3, ×4, ×8, ×16, and ×32, using PSNR as the evaluation metric. As shown in Fig. 11, HADF consistently outperforms all baselines across

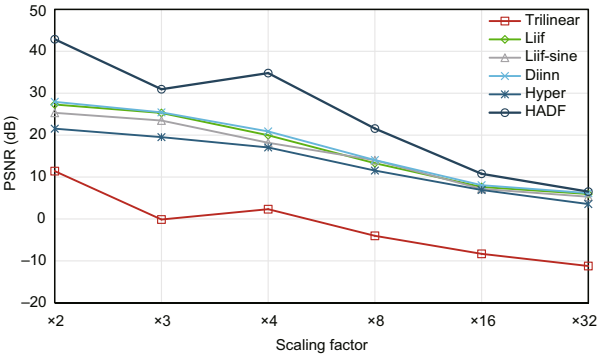

Fig. 11 PSNR performance comparison with 30% HR data unavailable

all scaling factors. By leveraging the LR consistency loss, HADF effectively uses LR data when HR data are partially unavailable, whereas baselines discard unpaired LR and HR data, reducing their training data. Consequently, HADF benefits from more data, significantly enhancing reconstruction quality.

The second experiment examines the robustness of the LR consistency loss to varying HR data unavailability rates. Fig. 12 illustrates the performance of HADF with or without the LR consistency loss, as the unavailability rate increases from 0.05, 0.10, 0.15, 0.20, 0.25, to 0.30; each PSNR value in Fig. 12 is the average of the PSNR values for a specific unavailability rate across all scaling factors. Without the loss,

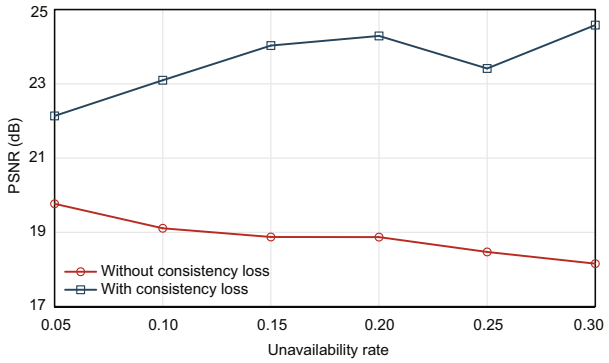

Fig. 12 Impact of the HR data unavailability rate on average PSNR

increasing the unavailability rate reduces available training data, leading to a steady decline in reconstruction quality. In contrast, HADF with the LR consistency loss addresses this issue by maintaining training data availability. Notably, HADF's PSNR even increases as the unavailability rate increases, because it considers the consistency between downsampled reconstructed HR data and LR data, incorporating more comprehensive reconstruction factors.

4.3.4 Noise robustness experiment

To further investigate the robustness of HADF under noisy conditions, we conduct an experiment focusing on the effect of noise in training data. For this purpose, we compare our proposed HADF model with Liif, which achieved the second-best performance in previous evaluations. The experiments are conducted using HIT data, with 80% of the dataset used for training, 10% for validation, and the remaining 10% for testing. For both models, we train one version using noise-free data and another using noisy data, and then evaluate their performance on noise-free test data. Gaussian noise is added to the inputs, with the noise magnitude randomly sampled from a broad range of 0.01–0.25 and scaled relative to the standard deviation of the data. This setup ensures that the models are exposed to perturbations of varying strength, covering both mild and severe noise scenarios.

As shown in Table 5, both HADF and Liif experience performance degradation when trained with noisy data, as reflected by the PSNR drop compared to their noise-free counterparts. However, HADF consistently demonstrates smaller PSNR drops across most scaling factors, indicating its stronger ability to maintain reconstruction accuracy under noisy training conditions. The only exception occurs at the $\times 32$ scale, where Liif shows a slightly smaller drop. Nevertheless, at this extreme upsampling factor, the overall PSNR values are already very low, leading to unstable results and diminishing the significance of this difference.

Overall, these results confirm that HADF exhibits clear noise robustness, outperforming existing models such as Liif in mitigating the negative impact of noisy training data. This robustness further highlights the practical value of HADF in real-world scenarios, where noise contamination in sparse sensor measurements is often unavoidable.

4.3.5 Ablation experiments

To investigate the influence of key parameters in hash-adaptive spatial encoding, we perform ablation experiments on the number of hierarchical levels L, the minimum layer resolution $N_{\min}$, and the maximum layer resolution $N_{\max}$. The PSNR performance across multiple upscaling factors ($\times 2$ to $\times 32$) for each parameter setting is summarized in Table 6.

From Table 6, we observe that $L = 3$ achieves slightly higher PSNR at $\times 3$ upscaling but yields lower values at almost all other scales compared with $L = 5$. The setting $L = 5$ achieves the highest overall performance, showing that this depth provides an effective balance between representation capacity and reconstruction accuracy. Increasing the depth further to $L = 7$ leads to lower PSNR across all scales, suggesting that additional layers do not enhance multi-scale representation and may introduce redundancy.

The setting $N_{\min} = 8$ produces generally lower PSNR compared with $N_{\min} = 16$, except for a marginal improvement at $\times 3$. Using $N_{\min} = 16$ achieves the highest PSNR across most scales, providing a sufficient coverage of large-scale structures. Increasing $N_{\min}$ to 32 results in reduced PSNR at all scales, indicating that an excessively dense coarse layer does not provide additional benefits.

The setting $N_{\max} = 256$ achieves lower PSNR results at nearly all scales compared with $N_{\max} = 512$. Increasing $N_{\max}$ to 512 yields the most balanced and consistently strong performance across scales. Further increasing to $N_{\max} = 1024$ causes performance degradation at all scales, suggesting redundancy and the lack of additional useful information at such HRs.

Table 5 Comparison of noise robustness across multiple scales (×2 to ×32) for models trained on noise-free or noisy data

Method	×2		×3		×4	
	PSNR (dB)	PSNR drop (%)	PSNR (dB)	PSNR drop (%)	PSNR (dB)	PSNR drop (%)
HADF without noise	50.7712	**8.59**	28.5964	**0.19**	41.2933	**4.68**
HADF with noise	46.4112		28.5415		39.3621	
Liif without noise	48.8186	12.75	28.5599	0.28	40.7151	8.54
Liif with noise	42.5946		28.4806		37.2377	
Method	×8		×16		×32	
	PSNR (dB)	PSNR drop (%)	PSNR (dB)	PSNR drop (%)	PSNR (dB)	PSNR drop (%)
HADF without noise	26.7773	**4.65**	16.0518	**13.83**	9.6162	19.00
HADF with noise	25.5329		13.8312		7.7892	
Liif without noise	26.5753	10.45	15.9334	19.88	9.6158	**17.55**
Liif with noise	23.7978		12.7658		7.9286	

PSNR drop value represents the degradation caused by noise, with the lower PSNR drop highlighted in bold

Table 6 PSNR performance across multiple scales (×2 to ×32) for ablation studies on multi-layer feature grid parameters

L^*	PSNR (dB)					
	×2	×3	×4	×8	×16	×32
3	50.5710	**28.6130**	41.0201	26.6594	16.0217	9.5798
5	**50.7712**	28.5964	**41.2933**	**26.7773**	**16.0518**	**9.6162**
7	49.6621	28.5857	41.1747	26.7437	16.0304	9.5893
$N_{\min}{}^{**}$	PSNR (dB)					
	×2	×3	×4	×8	×16	×32
8	50.5530	**28.5994**	41.0218	26.6630	16.0197	9.5836
16	**50.7712**	28.5964	**41.2933**	**26.7773**	**16.0518**	**9.6162**
32	50.4181	28.5959	41.0658	26.6755	16.0139	9.5641
$N_{\max}{}^{***}$	PSNR (dB)					
	×2	×3	×4	×8	×16	×32
256	**51.1281**	28.5914	41.0510	26.6634	16.0145	9.5743
512	50.7712	**28.5964**	**41.2933**	**26.7773**	**16.0518**	**9.6162**
1024	50.2021	28.5934	41.0179	26.6663	16.0229	9.5788

* $N_{\min}$=16, $N_{\max}$=512; ** L=5, $N_{\max}$=512; *** L=5, $N_{\min}$=16. The best PSNR and the selected optimal parameter combination are in bold

The ablation study demonstrates that the parameter combination of $L = 5$, $N_{\min} = 16$, and $N_{\max} = 512$ provides the best overall performance. The number of hierarchical levels $L = 5$ balances multi-scale representation depth and interpolation accuracy, the minimum resolution $N_{\min} = 16$ ensures effective coverage of large-scale structures, and the maximum resolution $N_{\max} = 512$ captures fine-scale features near the Kolmogorov scale. The hierarchical design allows the network to represent turbulent structures across scales from large integral motions to small-scale dissipative eddies, achieving both accurate reconstruction and high computational efficiency.

5 Conclusions

Turbulence is critical in fluid dynamics because HR data are essential for advancing scientific understanding and engineering applications. However, acquiring HR data presents numerous difficulties. To address these challenges, we have developed HADF, a novel network model incorporating hash adaptive spatial encoding, dynamic feature fusion, and an LR consistency loss. These innovations enable multi-scale turbulence reconstruction after training once and eliminate reliance on paired LR and HR data, supporting effective reconstruction even with partially available HR data. Experimental comparisons with representative baseline models cover interpolation, extrapolation, and unpaired data scenarios. HADF demonstrates superior global reconstruction

accuracy, preservation of local physical properties, precise recovery of fine turbulence details, and robustness to noise, highlighting its effectiveness across diverse reconstruction tasks and potential for practical turbulence modeling.

In addition, although the current work uses only DNS data to ensure a controlled testing environment, we recognize that fusing heterogeneous data sources, such as DNS and particle image velocimetry, could provide a more comprehensive framework for turbulence super-resolution. Recent work (Fukami and Taira, 2025) on observable-augmented manifold learning has shown that low-rank manifolds capturing turbulence structures across datasets of different origins can be extracted through nonlinear autoencoder-based compression. Incorporating similar multi-source data fusion strategies into HADF would likely enhance its robustness and applicability, offering a promising direction for future research.

Contributors

Yunfei LIU designed the research and drafted the paper. Gen ZHANG, Qingyang ZHANG, Qinglin WANG, and Jie LIU helped organize the paper. Xinhai CHEN revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

All datasets used in this study are publicly available at https://blastnet.github.io. The other data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Bode M, Gauding M, Lian ZY, et al., 2021. Using physics-informed enhanced super-resolution generative adversarial networks for subfilter modeling in turbulent reactive flows. *Proc Combust Inst*, 38(2):2617-2625. https://doi.org/10.1016/j.proci.2020.06.022

Brunton SL, Noack BR, Koumoutsakos P, 2020. Machine learning for fluid mechanics. *Annu Rev Fluid Mech*, 52(1):477-508. https://doi.org/10.1146/annurev-fluid-010719-060214

Cant RS, Ahmed U, Fang J, et al., 2022. An unstructured adaptive mesh refinement approach for computational fluid dynamics of reacting flows. *J Comput Phys*, 468:111480. https://doi.org/10.1016/j.jcp.2022.111480

Chen XH, Chen RL, Wan Q, et al., 2021a. An improved data-free surrogate model for solving partial differential equations using deep neural networks. *Sci Rep*, 11(1):19507. https://doi.org/10.1038/s41598-021-99037-x

Chen XH, Gong CY, Wan Q, et al., 2021b. Transfer learning for deep neural network-based partial differential equations solving. *Adv Aerodynam*, 3:36. https://doi.org/10.1186/s42774-021-00094-7

Chen XH, Wang ZC, Deng L, et al., 2024. Towards a new paradigm in intelligence-driven computational fluid dynamics simulations. *Eng Appl Comput Fluid Mech*, 18(1):2407005. https://doi.org/10.1080/19942060.2024.2407005

Chen YB, Liu SF, Wang XL, 2021. Learning continuous image representation with local implicit image function. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.8624-8634. https://doi.org/10.1109/CVPR46437.2021.00852

Chung WT, Mishra AA, Ihme M, 2022. Interpretable data-driven methods for subgrid-scale closure in LES for transcritical LOX/GCH4 combustion. *Combust Flame*, 239:111758. https://doi.org/10.1016/j.combustflame.2021.111758

Chung WT, Ihme M, Jung KS, et al., 2023a. BLASTNet simulation dataset (2.0). Zenodo. https://doi.org/10.5281/zenodo.8034232

Chung WT, Akoush B, Sharma P, et al., 2023b. Turbulence in focus: benchmarking scaling behavior of 3D volumetric super-resolution with BLASTNet 2.0 data. Proc 37th Int Conf on Neural Information Processing Systems, p.77430-77484.

Davidson P, 2015. Turbulence: an Introduction for Scientists and Engineers (2nd Ed.). Oxford University Press. https://doi.org/10.1093/acprof:oso/9780198722588.001.0001

Erichson NB, Mathelin L, Yao ZW, et al., 2020. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proc Roy Soc A*, 476(2238):20200097. https://doi.org/10.1098/rspa.2020.0097

Fukami K, Taira K, 2024. Single-snapshot machine learning for super-resolution of turbulence. *J Fluid Mech*, 1001:A32. https://doi.org/10.1017/jfm.2024.1136

Fukami K, Taira K, 2025. Observable-augmented manifold learning for multi-source turbulent flow data. *J Fluid Mech*, 1010:R4. https://doi.org/10.1017/jfm.2025.383

Fukami K, Fukagata K, Taira K, 2019. Super-resolution reconstruction of turbulent flows with machine learning. *J Fluid Mech*, 870:106-120. https://doi.org/10.1017/jfm.2019.238

Fukami K, Fukagata K, Taira K, 2023. Super-resolution analysis via machine learning: a survey for fluid flows. *Theor Comput Fluid Dynam*, 37(4):421-444. https://doi.org/10.1007/s00162-023-00663-0

Fukami K, Goto S, Taira K, 2024. Data-driven nonlinear turbulent flow scaling with Buckingham Pi variables. *J Fluid Mech*, 984:R4. https://doi.org/10.1017/jfm.2024.211

Jung KS, Kim SO, Lu TF, et al., 2021. On the flame stabilization of turbulent lifted hydrogen jet flames in heated coflows near the autoignition limit: a comparative DNS study. *Combust Flame*, 233:111584. https://doi.org/10.1016/j.combustflame.2021.111584

Liu HC, Sun B, Cai WW, 2019. kHz-rate volumetric flame imaging using a single camera. *Opt Commun*, 437:33-43. https://doi.org/10.1016/j.optcom.2018.12.036

Liu N, Ma L, 2020. Regularized tomographic PIV for incompressible flows based on conservation of mass. *Appl Opt*, 59(6):1667-1677. https://doi.org/10.1364/AO.380720

Morimoto M, Fukami K, Zhang K, et al., 2021. Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization. *Theor Comput Fluid Dynam*, 35(5):633-658. https://doi.org/10.1007/s00162-021-00580-0

Müller T, Evans A, Schied C, et al., 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans Graph*, 41(4):102. https://doi.org/10.1145/3528223.3530127

Nguyen QH, Beksi WJ, 2023. Single image super-resolution via a dual interactive implicit neural network. Proc IEEE/CVF Winter Conf on Applications of Computer Vision, p.4925-4934. https://doi.org/10.1109/WACV56688.2023.00491

Page J, 2025. Super-resolution of turbulence with dynamics in the loss. *J Fluid Mech*, 1002:R3. https://doi.org/10.1017/jfm.2024.1202

Pan SW, Brunton SL, Kutz JN, 2023. Neural implicit flow: a mesh-agnostic dimensionality reduction paradigm of spatio-temporal data. *J Mach Learn Res*, 24(1):1607-1666.

Pareja J, Johchi A, Li T, et al., 2019. A study of the spatial and temporal evolution of auto-ignition kernels using time-resolved tomographic OH-LIF. *Proc Combust Inst*, 37(2):1321-1328. https://doi.org/10.1016/j.proci.2018.06.028

Park K, Sinha U, Hedman P, et al., 2021a. HyperNerf: a higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans Graph*, 40(6):238. https://doi.org/10.1145/3478513.3480487

Park K, Sinha U, Barron JT, et al., 2021b. Nerfies: deformable neural radiance fields. Proc IEEE/CVF Int Conf on Computer Vision, p.5845-5854. https://doi.org/10.1109/ICCV48922.2021.00581

Savard B, Hawkes ER, Aditya K, et al., 2019. Regimes of premixed turbulent spontaneous ignition and deflagration under gas-turbine reheat combustion conditions. *Combust Flame*, 208:402-419. https://doi.org/10.1016/j.combustflame.2019.07.020

Shen LY, Pauly J, Xing L, et al., 2024. NeRP: implicit neural representation learning with prior embedding for sparsely sampled image reconstruction. *IEEE Trans Neur Netw Learn Syst*, 35(1):770-782. https://doi.org/110.1109/TNNLS.2022.3177134

Sitzmann V, Martel JNP, Bergman AW, et al., 2020. Implicit neural representations with periodic activation functions. Proc 34th Int Conf on Neural Information Processing Systems, p.7462-7473.

Wang B, Hui X, Li F, et al., 2022. Research status and key technology analysis of dynamic combustion in aeroengine main combustor. *J Aerosp Power*, 37(11):2479-2487 (in Chinese). https://doi.org/10.13224/j.cnki.jasp.20220223

Xu Q, Zhuang ZJ, Pan YC, et al., 2023. Super-resolution reconstruction of turbulent flows with a Transformer-based deep learning framework. *Phys Fluids*, 35(5):055130. https://doi.org/10.1063/5.0149551

Xu WJ, Luo WY, Wang Y, et al., 2020. Data-driven three-dimensional super-resolution imaging of a turbulent jet flame using a generative adversarial network. *Appl Opt*, 59(19):5729-5736. https://doi.org/10.1364/AO.392803

Yang Z, Yang H, Yin ZP, 2023. Super-resolution reconstruction for the three-dimensional turbulence flows with a back-projection network. *Phys Fluids*, 35(5):055123. https://doi.org/10.1063/5.0147902

Zhang KW, Zhu DD, Min XK, et al., 2023. Implicit neural representation learning for hyperspectral image super-resolution. *IEEE Trans Geosci Remote Sens*, 61:5500212. https://doi.org/10.1109/TGRS.2022.3230204