



RESEARCH ARTICLE | NOVEMBER 18 2024

A conditional deep learning model for super-resolution reconstruction of small-scale turbulent structures in particle-Laden flows ^{EP}

Hesam Tofghian   ; Jordan A. Denev  ; Nikolai Kornev 



Physics of Fluids 36, 115173 (2024)

<https://doi.org/10.1063/5.0235192>



Articles You May Be Interested In

Super-resolution reconstruction of turbulent velocity fields using a generative adversarial network-based artificial intelligence framework

Physics of Fluids (December 2019)

Flow field recovery in restricted domains using a generative adversarial network framework

Physics of Fluids (December 2024)

Three-dimensional ESRGAN for super-resolution reconstruction of turbulent flows with tricubic interpolation-based transfer learning

Physics of Fluids (December 2022)

17 December 2024 11:15:52



Physics of Fluids

Special Topics Open for Submissions

[Learn More](#)

A conditional deep learning model for super-resolution reconstruction of small-scale turbulent structures in particle-Laden flows

Cite as: Phys. Fluids **36**, 115173 (2024); doi: [10.1063/5.0235192](https://doi.org/10.1063/5.0235192)

Submitted: 26 August 2024 · Accepted: 29 October 2024 ·

Published Online: 18 November 2024



View Online



Export Citation



CrossMark

Hesam Tofighian,^{1,2,a)}  Jordan A. Denev,¹  and Nikolai Kornev² 

AFFILIATIONS

¹Scientific Computing Center, Department of Scientific Computing and Mathematics, Karlsruhe Institute of Technology, Karlsruhe 76131, Germany

²Chair of Modeling and Simulation, University of Rostock, Rostock 18051, Germany

^{a)}Author to whom correspondence should be addressed: hesam.tofighian@kit.edu

ABSTRACT

Super-resolution reconstruction of turbulent flows using deep learning has gained significant attention, yet challenges remain in accurately capturing physical small-scale structures. This study introduces the Conditional Enhanced Super-Resolution Generative Adversarial Network (CESRGAN) for reconstructing high-resolution turbulent velocity fields from low-resolution inputs. CESRGAN consists of a conditional discriminator and a conditional generator, the latter being called CoGEN. CoGEN incorporates subgrid-scale (SGS) turbulence kinetic energy as conditional information, improving the recovery of small-scale turbulent structures with the desired level of energy. By being aware of SGS turbulence kinetic energy, CoGEN is relatively insensitive to the degree of detail in the input. As shown in the paper, its advantages become more pronounced when the model is applied to heavily filtered input. We evaluate the model using direct numerical simulation (DNS) data of forced homogeneous isotropic turbulence. The analysis of Q-criterion isosurfaces, energy spectra, and probability density functions shows that the proposed CoGEN reconstructs fine-scale vortical structures more precisely and captures turbulent intermittency better compared to the traditional generator. Particle-pair dispersion simulations validate the physical fidelity of CoGEN-reconstructed fields, closely matching DNS results across various Stokes numbers and filtering levels. This paper demonstrates how incorporating available physical information enhances super-resolution models for turbulent flows.

© 2024 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0235192>

I. INTRODUCTION

Turbulent flows, characterized by their chaotic and multi-scale nature, play a crucial role in numerous engineering and scientific applications. The accurate representation of these flows across all scales, from the large energy-containing eddies to the small dissipative structures, is essential for understanding and predicting various phenomena such as particle dispersion, heat transfer, and mixing processes. Large-scale structures dominate the transport of momentum and energy, while small-scale structures are critical for processes like dissipation, fine-scale mixing, and dispersion. However, resolving all scales of turbulent motion poses significant computational challenges. While direct numerical simulation (DNS) provides the most accurate results by resolving all scales, its computational cost becomes prohibitive for many practical applications. Large-eddy simulation (LES) offers a compromise by resolving large-scale motions while filtering

out small-scale structures, approximating their effects through subgrid-scale (SGS) models. This limitation is particularly problematic in particle-laden flows, where small-scale turbulent structures significantly influence particle behavior.^{1,2}

Lagrangian stochastic models have been widely employed to account for subgrid-scale turbulence effects on particle dispersion.^{3–6} However, these models often struggle to accurately predict pair statistics. This limitation arises because they generate velocity fluctuations independently for each particle, thus neglecting the spatial correlations inherent in turbulent flows. To address these challenges, researchers have developed various innovative methods to reconstruct high-resolution turbulent flow fields from lower-resolution data. For example, in the approximate deconvolution method (ADM),^{7,8} the velocity field is enriched with small-scale structures by applying an inverse filtering operation to the resolved large-scale field. However, this method

only reconstructs scales near the filter cutoff, neglecting smaller subgrid motions. Building on these approaches, Bassenne *et al.*⁹ introduced the spectrally enriched differential filter (SDF) model. This method combines ADM with a dynamic procedure to reconstruct velocity fields containing scales smaller than the LES grid resolution. The main problem of this model is that the generated velocity field is not inherently divergence-free, thus requiring a computationally expensive divergence-free projection step. Fractal interpolation (FI) is another sub-filter scale reconstruction method, based on turbulence fractality assumption.^{10,11} The effectiveness of FI relies on accurate determination of fractal dimensions, which vary across flow regimes. Moreover, a major limitation emerges in 3D reconstructions, where FI generates uncorrelated velocity components. Kinematic simulation (KS) is another reconstruction model that generates SGS velocity field by superposing several random orthogonal Fourier modes.^{12,13} However, these global Fourier basis functions have infinite support in physical space, restricting KS's application primarily to homogeneous turbulence. In contrast, the Gabor mode reconstruction method provides a more localized approach by employing spectrally localized Gabor modes.^{14,15} The SGS velocity field can also be reconstructed by utilizing wavelet-based methods. Hausmann *et al.*¹⁶ proposed a wavelet enrichment model that discretizes the SGS velocity using a divergence-free wavelet vector basis, enabling the generation of inhomogeneous and anisotropic subgrid-scale velocity fields. Although these methods can predict certain turbulence statistics, they often generate velocity fields that deviate significantly from DNS results and introduce numerical artifacts. These limitations have motivated researchers to explore data-driven approaches for turbulence reconstruction.

Concurrently, the computer vision community has developed deep-learning based methods for super-resolution tasks, a process of enhancing low-resolution images to high resolution. Dong *et al.*¹⁷ pioneered the application of deep convolutional neural networks (CNNs) to super-resolution by introducing the Super-Resolution Convolutional Neural Network (SRCNN) architecture. SRCNN comprised only three convolutional layers but demonstrated remarkable improvements over traditional super-resolution methods. Since then, there have been many advancements in CNNs-based super-resolution to improve the reconstruction accuracy and training efficiency.^{18–22} Despite these advancements, a common limitation of many super-resolution methods is their focus on minimizing pixel-wise loss functions, which results in overly smooth textures that lack the high-frequency details. To address this issue, Ledig *et al.*²³ introduced Super-Resolution GAN (SRGAN), applying the concept of Generative Adversarial Networks (GANs) to super-resolution tasks. SRGAN builds upon the GAN framework originally proposed by Goodfellow *et al.*,²⁴ utilizing a generator network to produce high-resolution images and a discriminator network to distinguish between real and generated images. This adversarial approach enables SRGAN to reconstruct images with more realistic textures and sharper details. Since then, several modifications have been made to SRGAN to improve output quality.²⁵ In this paper, we use Enhanced SRGAN (ESRGAN)²⁶ as the base model for our development. ESRGAN, which is an improved version of SRGAN's architecture, incorporates residual-in-residual dense blocks and a relativistic adversarial loss²⁷ for enhancing perceptual quality and stable training.

Recently, researchers have begun applying deep learning SR techniques in turbulent flow reconstruction. Fukami *et al.*²⁸ pioneered the application of deep learning for the reconstruction of turbulent flows.

They developed two CNN-based SR models including a simple CNN inspired by Dong *et al.*¹⁷ and a customized CNN model called hybrid downsampled skip-connection/multi-scale (DSC/MS). Their approach accurately reconstructed 2D isotropic turbulence, outperforming traditional interpolation methods in capturing small-scale structures. They further extended the application of the hybrid DSC/MS model to 3D turbulent channel flow.²⁹ Liu *et al.*³⁰ also applied CNNs to reconstruct 2D slices of 3D turbulent flows, showing effective performance in both isotropic (forced turbulence) and anisotropic (channel flow) cases. Zhou *et al.*³¹ further advanced CNN approach by developing a turbulence volumetric super-resolution (TVSR) model that directly processes 3D flow field data using a patchwise training strategy. This approach demonstrated good generalization capability across different Reynolds numbers. Subramaniam *et al.*³² have developed a physics-informed SRGAN that incorporate the continuity equation in the loss function as a physical constraint. They compared their SRGAN model with a CNN-based model, showing that the SRGAN outperformed the CNN-based model in minimizing the physics loss. Kim *et al.*³³ introduced an unsupervised CycleGAN model that successfully reconstructs high-resolution 2D turbulent fields from unpaired data, broadening the applicability of super-resolution turbulent flow reconstruction. Among the diverse versions of GAN-based approaches for super-resolution, ESRGAN is extensively employed for reconstructing turbulent flow fields due to its accuracy in capturing small-scale structures.^{34–41} Bode *et al.*³⁴ developed a Physics-Informed ESRGAN (PIESRGAN) based on the ESRGAN architecture, applying it to complex turbulent reactive flows.^{36,39,41} They demonstrated that their PIESRGAN-LES approach achieved approximately a 9.6 times speedup for the inference step compared to DNS.⁴¹ PIESRGAN has also been tested for interfacial flows, accurately reconstructing volume-of-fluid fields in multiphase simulations.⁴² Nista *et al.*⁴⁰ compared a supervised deep-CNN with a semisupervised/unsupervised PIESRGAN, showing that the PIESRGAN outperformed the CNN-based model, particularly in reconstructing small-scale structures and generalizing to out-of-sample conditions. Specifically, they demonstrate that their ESRGAN model effectively extrapolates to out-of-sample flows, particularly at higher Reynolds numbers. Deng *et al.*³⁵ demonstrated ESRGAN's superior turbulence reconstruction performance over SRGAN in both instantaneous and mean flow fields. ESRGAN architecture was also used by Yousif *et al.*⁴³ in their multi-scale model (MS-ESRGAN) for 2D turbulent channel flows at different Reynolds numbers. Yu *et al.*³⁷ extended the MS-ESRGAN to three dimensions, developing a D-ESRGAN model that incorporates transfer learning techniques and physics-based loss functions. ESRGAN has also shown promise in industrial applications. Recent work by Trinh *et al.*⁴⁴ demonstrated successful application of super-resolution models to vehicle aerodynamics, enabling enrichment of flow fields around vehicle geometries.

In this work, we propose a novel conditional super-resolution model for turbulent flows based on the ESRGAN architecture. Our Conditional-ESRGAN (CESRGAN) consists of a conditional generator (CoGEN) and a conditional discriminator. Unlike existing super-resolution methods that typically do not utilize SGS information due to their origins in computer vision tasks, our CoGEN is conditioned on SGS turbulence kinetic energy (k_{SGS}). This k_{SGS} information, available in most scale-resolving simulations, enables CoGEN to reconstruct high-resolution turbulent velocity fields with the desired level of

turbulence kinetic energy. By incorporating k_{SGS} , our CoGEN can effectively handle inputs with different degrees of smoothness, addressing a common challenge in turbulent flow reconstruction. Our results demonstrate that CoGEN outperforms traditional generators (TradGEN), particularly in capturing small-scale turbulent structures crucial for particle-laden flow simulations. To show the model's application in particle-laden flow simulations, we integrate the trained generators with a CFD solver to simulate particle-pair dispersion in turbulent flows. Our CESRGAN not only improves the accuracy of turbulence reconstruction but also enhances GAN training stability by incorporating a conditional discriminator.

The remainder of this paper is organized as follows: Sec. II describes the methodology, including the details of our CESRGAN architecture, data preparation, and training process. Section III presents the results and discussion, focusing on the reconstruction of vortex structures, turbulent flow statistics, and dispersed-phase statistics. Finally, Sec. IV concludes the paper.

II. METHODOLOGY

A. Deep learning model for super-resolution

In this study, our model builds upon the ESRGAN architecture,²⁶ which has shown promising results in reconstructing small-scale structures in turbulent flows.^{40,41} The ESRGAN is a GAN-type architecture that consists of two adversarial neural networks: a generator (G) and a discriminator (D). In a standard GAN, the generator typically produces synthetic data samples, while the discriminator evaluates the authenticity of these samples compared to real data.²⁴ For super-resolution tasks, this framework is adapted so that the generator specifically learns to produce high-resolution data from low-resolution inputs.²³ The network G and D are trained in a two-player min-max game process with the objectives formulated as follows:

$$\min_G \max_D \left\{ \mathbb{E}_{Y_r \sim P_{Y_r}} [\log(D_{Ra}(Y_r, Y_g))] \right. \\ \left. + \mathbb{E}_{Y_g \sim P_{Y_g}} [\log(1 - D_{Ra}(Y_g, Y_r))] \right\}, \quad (1)$$

where \mathbb{E} denotes the expectation operator. Y_r refers to a sample set drawn from the real high-resolution (HR) data distribution (P_{Y_r}) and $Y_g = G(X)$ denotes a sample set obtained from the generated high-resolution data distribution (P_{Y_g}), where X is a sample set of low-resolution (LR) inputs. In Eq. (1), D_{Ra} represents the relativistic discriminator,²⁷ which is defined as

$$D_{Ra}(Y_1, Y_2) = \sigma \left(D(Y_1) - \mathbb{E}_{Y_2 \sim P_{Y_2}} [D(Y_2)] \right), \quad (2)$$

where σ represents the sigmoid function and $D(Y)$ is the output of the discriminator. $D_{Ra}(Y_1, Y_2)$ provides a score ranging from 0 to 1, indicating the relative authenticity of sample set Y_1 compared to sample set Y_2 . In Eq. (2), Y_1 can be a sample set from either P_{Y_r} or P_{Y_g} , with Y_2 then representing a sample set from the other distribution. The choice of which distribution is assigned to Y_1 depends on whether we are evaluating real or generated data. Specifically, $D_{Ra}(Y_g, Y_r)$ returns a score indicating how much more realistic Y_g is compared to Y_r , while $D_{Ra}(Y_r, Y_g)$ provides a score showing how much more realistic Y_r is compared to Y_g . Throughout the training process, the trainable parameters of G are updated to minimize Eq. (1). This optimization leads to an increase in the relativistic score of the generated data, i.e., $D_{Ra}(Y_g, Y_r)$,

signifying an improvement in its capability to upscale low-resolution data to realistic high-resolution data. Simultaneously, the parameters of D are adjusted to maximize Eq. (1), enabling it to assign lower scores to generated data and higher scores to real data, represented, respectively, as lower $D_{Ra}(Y_g, Y_r)$ and higher $D_{Ra}(Y_r, Y_g)$. The purpose of this adversarial framework is to improve both networks through competition. At the end of training, G produces the reconstructed high-resolution data, closely resembling real data.

1. Architecture

Here, we propose a novel conditional variant of ESRGAN (CESRGAN) tailored specifically for the task of learning the mapping function from LR to HR turbulent velocity fields. The key difference between CESRGAN and the traditional ESRGAN lies in its conditional inputs. In CESRGAN, the generator is conditioned on the k_{SGS} , while the discriminator is conditioned on the LR velocity field. Further details on these conditional inputs are provided in Sec. II A 2.

The detailed structure of CESRGAN is sketched in Figs. 1 and 2. Both the generator and discriminator mainly consist of convolutional layers, employing a kernel size of 3^3 for their convolutional operations. The backbone of generator is Residual in Residual Dense Block (RRDB),²⁶ a deep CNN unit comprising multiple Residual Dense Blocks (RDB)⁴⁵ linked together via the residual scaling factor of $\beta = 0.2$. Here, the generator is structured with four stacked RRDBs, with each RRDB containing three embedded RDBs as shown in Fig. 1. Each RDB is composed of convolutional layers paired with leaky rectified linear unit (LReLU) activation functions and incorporates dense skip connections. These connections facilitate the flow of information across different layers, aiding in the extraction and preservation of fine details during the super-resolution process and avoiding the gradient vanishing or exploding.^{23,46} Following the RRDBs, upsampling is done through two interpolation blocks with scaling factor of 2, resulting in a total increase in resolution by a factor of 4 in each direction.

The architecture of the discriminator, as depicted in Fig. 2, comprises multiple stacked layers of CNNs, Leaky Rectified Linear Unit (LReLU) activation functions, and Batch Normalization (BN). Toward the end of this architecture, a Flattening operation is executed to transition from Convolutional Layers to Fully Connected Layers. This step is done as it transforms the multidimensional output of the convolutional layers into the single scalar output. Furthermore, a dropout block is integrated into the architecture based on recommendations from Bode *et al.*³⁶ for preventing the discriminator from overfitting.

2. Proposed conditional networks

Super-resolution is an ill-posed problem, which means a low-resolution input lacks sufficient information to uniquely determine a corresponding ground truth high-resolution output.⁴⁷ This uncertainty is increased when the smoothness level of low-resolution data varies across different data samples.⁴⁸ In LES, the smoothness level is correlated with the k_{SGS} , which either is directly available in the turbulence model or can be estimated. In this study, this extra information is used to alleviate the ill-posedness nature of super-resolution. As illustrated in Fig. 1, the generator is conditioned on k_{SGS} by concatenating it with the LR velocity field, forming the input for the network. The conditional generator's output can be expressed mathematically as $Y_g = G(X|k_{SGS})$. In the proposed conditional model, the generator is

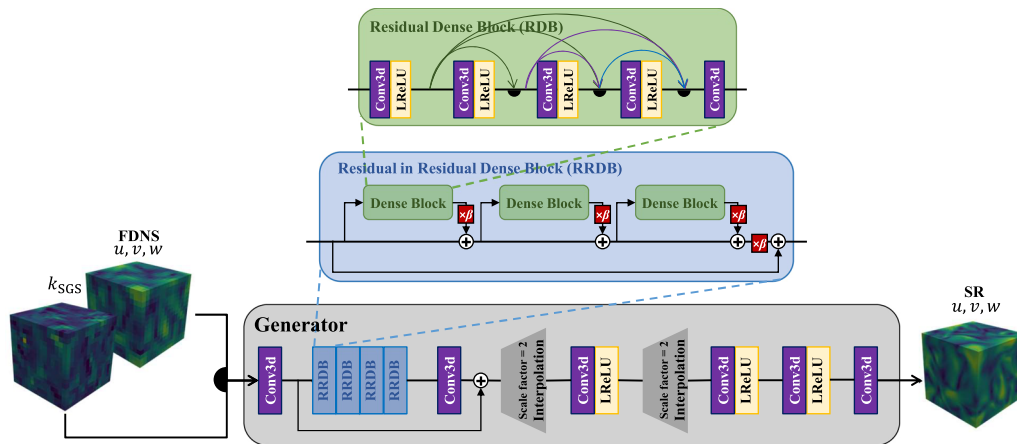


FIG. 1. Detailed architecture of the proposed conditional generator (CoGEN). The generator takes filtered DNS (FDNS) velocity components (u, v, w) and k_{SGS} as inputs to produce super-resolution (SR) velocity fields. The \oplus symbol denotes element-wise summation, while filled half-circle symbol represents concatenation operations.

forced to learn the correlation between k_{SGS} and the smoothness level of X velocity field to avoid retraining whenever the level of smoothing changes. As a result, the generator can effectively handle LR data with varying smoothness levels, thereby enhancing the model’s generalizability.

In addition to utilizing a conditional generator, a conditional discriminator is also employed. The traditional discriminator merely judges whether the HR data conforms to the real distribution, neglecting to assess its fidelity to the original LR input. While the pixel-wise content loss aims to increase fidelity to the ground truth data, the adversarial loss drives the generator to create any plausible realistic output, which may not necessarily align with the ground truth data. This inconsistency between pixel-wise content loss and adversarial loss often leads to the emergence of artifacts in the reconstructed data²³ and contributes to training instability.⁴⁹

To address these problems, we employ a conditional discriminator, inspired by the approach introduced by Zhang *et al.*⁴⁹ In this method both LR and HR data are provided to the discriminator network, with LR data acting as the conditioning information. As illustrated in Fig. 2, the LR data are upsampled through nearest neighbors interpolation to align with the size of HR data and then concatenated with it before being fed to the discriminator. By conditioning the discriminator on the LR input, we ensure that the discriminator incorporates LR data into its assessment. We extend this conditional framework by modifying the discriminator’s objective function based on the suggestion from Yin.⁵⁰ He suggests to penalize the discriminator when it assigns a high score to HR data that lacks correlation with

the corresponding LR data, even if the HR data are sampled from the ground truth dataset. To formulate the new objective function in conditional mode, we modify Eq. (1) as follows:

$$\min_G \max_D \left\{ \mathbb{E}_{Y_r \sim P_{Y_r}} [\log (D_{Ra}(Y_r, Y_g|X))] + \mathbb{E}_{Y_g \sim P_{Y_g}} [\log (1 - D_{Ra}(Y_g, Y_r|X))] + \mathbb{E}_{Y'_g \sim P_{Y_g}} [\log (1 - D_{Ra}(Y'_g, Y_g|X))] \right\}, \quad (3)$$

where Y'_r is a shuffled version of Y_r , intended to mismatch with the X . The last term of Eq. (3) forces discriminator to give a low score to the mismatched inputs. Incorporating this term additionally facilitates the alignment of adversarial loss with pixel-wise content loss, leading to a more stable training. The conditional relativistic discriminator is then defined as follows:

$$D_{Ra}(Y_1, Y_2|X) = \sigma \left(D(Y_1|X) - \mathbb{E}_{Y_2 \sim P_{Y_2}} [D(Y_2|X)] \right). \quad (4)$$

Here, $D(Y|X)$ represents a conditional discriminator network, with the condition of X .

3. Loss functions

The training of our CESRGAN model involves optimizing separate loss functions for the generator and discriminator networks.

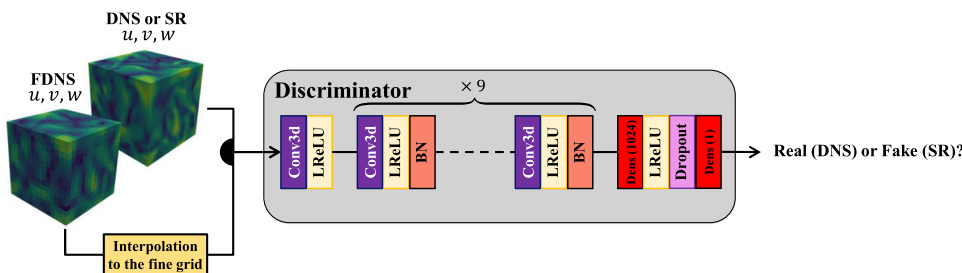


FIG. 2. Detailed architecture of the proposed conditional discriminator. The network takes both filtered DNS (FDNS) and either DNS or super-resolution (SR) velocity fields as inputs. It classifies the input as real (DNS) or fake (SR). The filled half-circle symbol represents concatenation operations.

a. Discriminator loss. The discriminator loss is derived from Eq. (3) and consists only of the adversarial component:

$$\begin{aligned} \mathcal{L}_D = & -\mathbb{E}_{Y_r \sim P_{Y_r}} [\log(D_{Ra}(Y_r, Y_g|X))] \\ & -\mathbb{E}_{Y_g \sim P_{Y_g}} [\log(1 - D_{Ra}(Y_g, Y_r|X))] \\ & -\mathbb{E}_{Y'_r \sim P_{Y'_r}} [\log(1 - D_{Ra}(Y'_r, Y_g|X))]. \end{aligned} \quad (5)$$

b. Generator loss. Similar to Bode *et al.*,³⁶ we train the generator with a loss function that combines adversarial, content, and physics-based components

$$\mathcal{L}_G = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{vel}\mathcal{L}_{vel} + \lambda_{grad}\mathcal{L}_{grad} + \lambda_{phys}\mathcal{L}_{phys}, \quad (6)$$

where: $\mathcal{L}_{adv} = -\mathbb{E}_{Y_g \sim P_{Y_g}} [\log(D_{Ra}(Y_g, Y_r|X))]$ is the adversarial loss derived from Eq. (3), $\mathcal{L}_{vel} = \frac{1}{N} \sum_{i=1}^N |Y_g^i - Y_r^i|_2^2$ is the velocity field pixel-wise loss, $\mathcal{L}_{grad} = \frac{1}{N} \sum_{i=1}^N |\nabla Y_g^i - \nabla Y_r^i|_2^2$ is the velocity gradient field pixel-wise loss, and $\mathcal{L}_{phys} = \frac{1}{N} \sum_{i=1}^N |\nabla \cdot Y_g^i|_2^2$ is the physical consistency loss based on the divergence-free condition for incompressible flows.

The choice of weighting parameters in the loss function affects the relative importance of different criteria and thereby influences the model's outcome. Since finding optimal hyperparameters requires extensive computational resources, we adopted most weighting parameters from Bode *et al.*³⁶ The selected values fall within a range shown to provide effective network performance for turbulence reconstruction tasks.³⁶ To further optimize training, we increased λ_{adv} to 10^{-4} , which accelerated convergence without adversely affecting the final content loss. The final values used in this work are $\lambda_{adv} = 10^{-4}$, $\lambda_{vel} = 0.89$, $\lambda_{grad} = 0.085$, and $\lambda_{phys} = 0.025$, which balance the contribution of each loss component.

B. Data preparation

1. Direct numerical simulation

In the context of super-resolution reconstruction for turbulent flow, a pair of DNS and filtered DNS (FDNS) data usually serves as the training dataset for the networks. In this study, the three-dimensional velocity field of a forced, homogeneous, isotropic turbulent flow is chosen as the basis for training and testing. This particular case has been extensively employed both for assessing super-resolution methods and for investigating the particle-turbulence interactions, rendering it well-suited for this study. The flow field data are generated using a DNS simulation implemented in the open-source OpenFOAM[®] v2112 packages by solving the following incompressible Navier–Stokes equations:

$$\nabla \cdot \mathbf{u} = 0, \quad (7)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{f}(\mathbf{x}, t). \quad (8)$$

Here, \mathbf{u} , t , p , ρ , ν , and $\mathbf{f}(\mathbf{x}, t)$ represent the velocity vector, time, pressure, fluid density, fluid kinematic viscosity, and stochastic forcing term, respectively. The computational domain is a periodic cubic box with an edge length of $2\pi(m)$, discretized using Cartesian uniform grids with a resolution of 64^3 . To minimize numerical dissipation, careful consideration must be given to the selection of discretization

schemes in DNS simulations. In this study, the diffusion and convection terms are discretized using the standard second-order and fourth-order central differencing schemes, respectively. For handling velocity-pressure coupling, the Pressure Implicit with Splitting of Operators (PISO) algorithm is employed. The Second-Order Upwind Euler (SOUE) scheme is utilized for discretizing time derivatives, while the maximum Courant–Friedrichs–Lewy number (CFL) is maintained below 0.2 during the simulation for numerical stability.

The Uhlenbeck–Ornstein (UO) random process-based forcing term $\mathbf{f}(\mathbf{x}, t)$ in Eq. (8) is utilized to sustain statistically stationary turbulence within the computational domain.⁵¹ This is achieved by continuously injecting kinetic energy solely into the low-wavenumber modes in the Fourier space. Since the force term operates primarily on the large scales, an auxiliary coarser mesh is enough to generate the force field, following a similar approach outlined in Ref. 52. Consequently, parallel Fast Fourier Transform (FFT) is unnecessary on the main fine mesh, and instead, the force field is generated on the auxiliary mesh with a resolution of 16^3 using serial FFT calculations. This inexpensive calculation is processed solely on the master CPU core, and upon generating the force field on the auxiliary mesh, different parts of the field are communicated to the associated worker CPU cores using the message passing interface (MPI). Subsequently, each CPU core interpolates its respective portion of the force field onto the main fine mesh. The flow statistics at the stationary state are listed in Table I.

2. Filtering and downsampling

To prepare the LR data for the super-resolution training, we generate the FDNS data by applying a discrete Gaussian filter to the original DNS data. The Gaussian filter kernel is defined in three dimensions as follows:

$$G(i\Delta, j\Delta, k\Delta) = \frac{1}{(2\pi\sigma^2)^{3/2}} \exp\left(-\frac{i^2 + j^2 + k^2}{2(\sigma/\Delta)^2}\right), \quad (9)$$

where i , j , and k are integer indices representing the number of grid cells away from the center of the Gaussian filter in each direction, Δ is the cell size of DNS grid, and σ is the standard deviation of the Gaussian distribution, determining the extent of the filter's spread and the degree of smoothing applied to the data. In our implementation, we use a filter kernel with a size of $13 \times 13 \times 13$ cells resulting i, j , and k ranging from -6 to 6 . To ensure that the total sum of the discrete Gaussian kernel over all space equals 1, we normalize the filter weights as

TABLE I. Statistics of the stationary turbulence.

Kolmogorov length scale (η)	0.0646 (m)
Kolmogorov time scale (τ_η)	0.208 (s)
Integral length scale (l)	2.415 (m)
Large eddy turnover time (t_l)	2.925 (s)
Taylor microscale (λ)	0.704 (m)
Taylor Reynolds number (Re_λ)	36.457 (–)
Turbulence kinetic energy (TKE)	1.138 (m^2/s^2)
Dissipation rate (ε)	0.458 (m^2/s^3)

$$G'(i\Delta, j\Delta, k\Delta) = \frac{G(i\Delta, j\Delta, k\Delta)}{\sum_{i=-6}^6 \sum_{j=-6}^6 \sum_{k=-6}^6 G(i\Delta, j\Delta, k\Delta)}. \quad (10)$$

The filtered velocity field $\tilde{\mathbf{u}}(x, y, z)$ is then obtained by convolving the DNS velocity field $\mathbf{u}(x, y, z)$ with the normalized discrete Gaussian filter,

$$\tilde{\mathbf{u}}(x, y, z) = \sum_{i=-6}^6 \sum_{j=-6}^6 \sum_{k=-6}^6 G'(i\Delta, j\Delta, k\Delta) \mathbf{u}(x + i\Delta, y + j\Delta, z + k\Delta). \quad (11)$$

The filtered field is then downsampled using a stride of 4, reducing the resolution from 64^3 to 16^3 . This two-step approach, filtering followed by downsampling, separates the smoothing process (controlled by σ) from the resolution reduction (achieved by the stride). It simulates the effect of a coarse-grid simulation while allowing for more flexibility in generating training data with various levels of detail loss. After filtering and downsampling, we calculate the k_{SGS} for each filtered sample using the difference between the DNS velocity field and the FDNS velocity field. This k_{SGS} value serves as the conditioning information for the conditional generator, providing crucial information about the level of detail lost during the filtering process.

To demonstrate the extent to which the filtering process removes small-scale details from the original DNS field, the FDNS data are compared with associated DNS data and reconstructed field in Fig. 3. Here, $\sigma = 4\Delta$ is selected for demonstration. However, to enhance the network's ability to handle different levels of filtering, we employ a variable standard deviation σ to generate the data for training. The value of σ is randomly generated between Δ and 5Δ for each training sample, creating a diverse set of filtered inputs. This approach allows the network to learn to reconstruct high-resolution fields from a wide range of smoothing intensities.

3. Training

The dataset for training the neural networks is created by pairing the FDNS data (input) with the corresponding DNS data (target output). We extract 6000 snapshots from the statistically stationary state of the simulation, with a time interval of $2t_l$ between consecutive snapshots. To prevent the overfitting, we apply random rotations and reflections to the velocity field as a data augmentation technique. Prior to feeding the data into the networks, we normalize each FDNS-DNS

pair using the root mean square (RMS) of the FDNS snapshot's velocity fluctuations. This snapshot-specific normalization approach differs from common normalization method,^{3,6,8} where FDNS and DNS data are normalized separately based on their respective global quantities. We chose this strategy for its practicality in real-world applications, where only a low-resolution snapshot is available at each time step during a new simulation, without access to global information from the entire simulation.

The deep neural networks were implemented using PyTorch. For training, we utilized the Adam optimizer with a fixed learning rate of 10^{-4} and a batch size of 40. Unlike previous studies,^{32,33,40} our model does not need adaptive learning rates or pre-training phases to achieve stable GAN training. This stability is achieved through the use of our conditional discriminator, which aligns the adversarial loss with the pixel-wise content loss, as detailed in Sec. II A 2. This conditional approach decreases training instabilities commonly encountered in GAN-based super-resolution models.

The computational resources for this study consisted of one node of the HoreKa supercomputer at the Karlsruhe Institute of Technology (KIT). Each node is equipped with four NVIDIA A100 GPUs. The total training time for the model was approximately 8 hours on this hardware configuration.

4. Testing

To evaluate the trained generator, we integrate it with OpenFOAM using the C++ API of PyTorch. This integration allows for real-time super-resolution reconstruction during the simulation process. At each time step, the DNS field and its corresponding FDNS field are obtained directly from OpenFOAM. The generator then uses the FDNS field to reconstruct a super-resolution (SR) velocity field on-the-fly. For testing purposes, we focus on two specific filtering levels: $\sigma = 2\Delta$ and $\sigma = 4\Delta$, representing moderate and strong filtering intensities, respectively. This allows us to evaluate the model's performance across different levels of detail loss. Table II details the testing configurations and naming conventions used in our study.

III. RESULTS AND DISCUSSION

A. Vortex structures reconstruction

To evaluate the effect of the conditional information, k_{SGS} , on the recovery of vortex structures, we computed the vortex criterion Q . The Q -criterion is the second invariant of the velocity gradient tensor and defined as

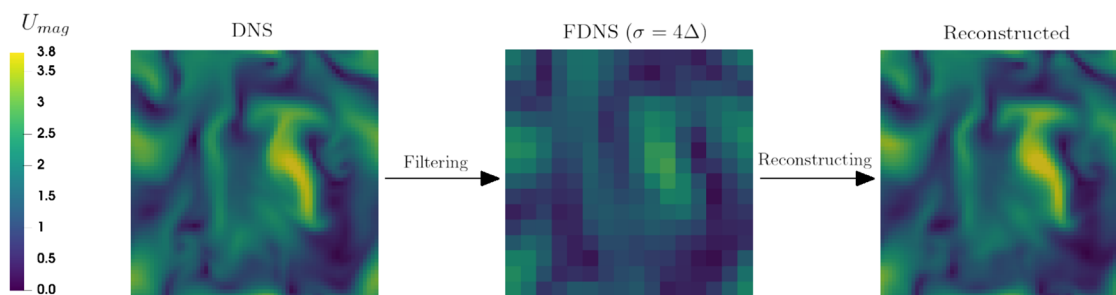


FIG. 3. Illustration of the filtering process and reconstruction. Left: Original DNS velocity magnitude field showing detailed turbulent structures. Center: FDNS field with $\sigma = 4\Delta$, demonstrating the removal of small-scale details due to filtering. Right: Reconstructed high-resolution field from the FDNS input.

TABLE II. SR testing configurations and naming conventions for filtered DNS (FDNS), traditional generator (TradGEN), and conditional generator (CoGEN) outputs at different filter levels.

Filter level	Description	Data type	Name
$\sigma = 2\Delta$	Moderate filtering (typical coarse simulation)	Filtered DNS (FDNS)	FDNS-2
		TradGEN output	TradGEN-2
		CoGEN output	CoGEN-2
$\sigma = 4\Delta$	Strong filtering (challenges reconstruction capabilities)	Filtered DNS (FDNS)	FDNS-4
		TradGEN output	TradGEN-4
		CoGEN output	CoGEN-4

$$Q = \frac{1}{2} [(\text{tr}(\nabla\mathbf{u}))^2 - \text{tr}(\nabla\mathbf{u} \cdot \nabla\mathbf{u})], \quad (12)$$

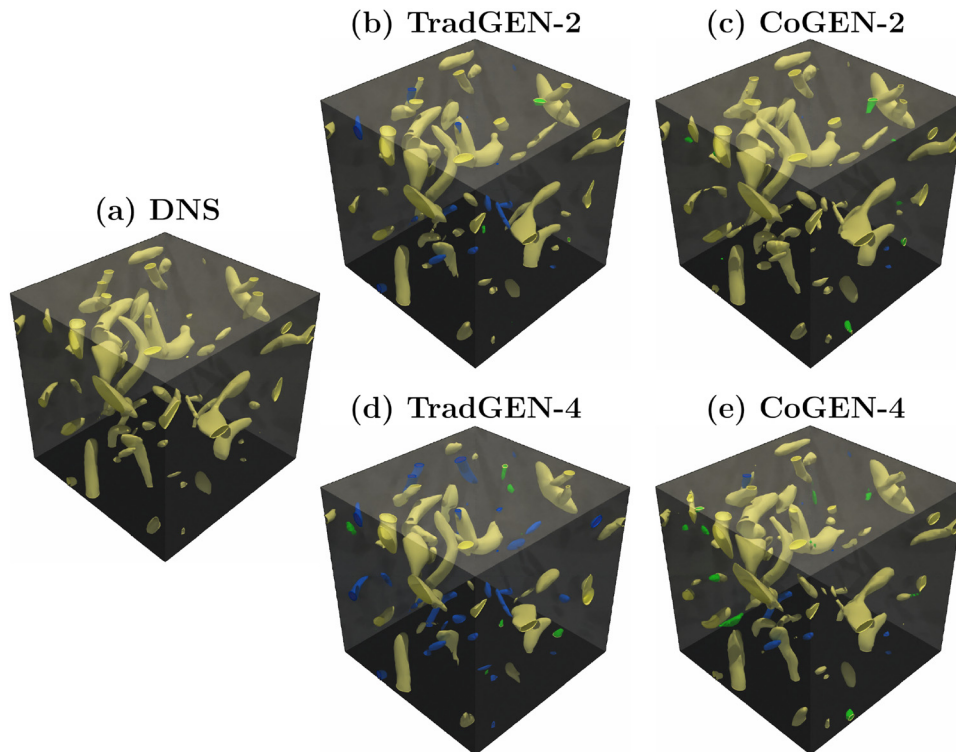
where $\nabla\mathbf{u}$ is the velocity gradient tensor, and tr denotes the trace of a matrix.

To visualize and compare the vortex structures across different reconstructed fields, we plotted isosurfaces of $Q = 0.86\tau_\eta^2$ in Fig. 4. Figure 4(a) displays the Q isosurfaces obtained from DNS, shown in yellow. Figures 4(b) and 4(c) display the Q isosurfaces from the reconstructed velocity fields by the TradGEN and CoGEN, respectively, for $\sigma = 2\Delta$. Figures 4(d) and 4(e) show the same for $\sigma = 4\Delta$. In these figures, the reconstructed vortex structures that match the DNS results are in yellow, those not recovered by the models and omitted are in blue, and those generated by the models but differing from the DNS structures are in green.

It is evident that both the conditional and traditional generators accurately reconstruct large vortex structures, regardless of the

smoothness level in the FDNS input. When it comes to small-scale structures, the conditional generator shows two advantages over the traditional generator.

First, the conditional generator demonstrates superior performance in fine structure recovery. A comparison of Figs. 4(b) and 4(c) reveals that although both TradGEN-2 and CoGEN-2 are generated with the same input (FDNS-2), the CoGEN was able to recover very fine structures, whereas the TradGEN missed several of these fine structures, which are shown in blue. A similar pattern can be observed by comparing Figs. 4(d) and 4(e), which are generated with the more heavily filtered input (FDNS-4). In both cases, the CoGEN preserves more of the original fine-scale vortical structures than the TradGEN. This discrepancy in the small-scale structures is a consequence of the ill-posed nature of the super-resolution reconstruction process, where multiple high-resolution reconstructions can exist for a given low-resolution input. In the CoGEN, this ill-posedness is alleviated by using


FIG. 4. Isosurfaces of the Q -criterion obtained from (a) DNS, (b) TradGEN-2, (c) CoGEN-2, (d) TradGEN-4, and (e) CoGEN-4 (as defined in Table II). Yellow structures are accurately reconstructed from the DNS, blue structures are omitted by the super-resolution models, and green structures are generated by the models but differ from the DNS.

k_{SGS} as additional information, constraining the range of plausible solutions. The TradGEN, lacking this constraint, encounters a wider range of plausible solutions. As a result, the pixel-wise loss guides the network toward learning an average of potential reconstructions, resulting in an overly smooth velocity field devoid of fine structures.⁵³

Second, the CoGEN shows an ability to compensate for omitted structures by generating new vortical structures, shown in green in Figs. 4(c) and 4(e). This capability is particularly evident in Fig. 4(e), where the input is more heavily filtered (CoGEN-4). When the FDNS velocity field lacks sufficient details to reconstruct all vortices exactly as they appear in the DNS, the CoGEN generates these new structures. While these newly generated vortical structures differ from those in the DNS, they serve to compensate for the lost vortical structures. This compensatory effect is less pronounced in the TradGEN results, as seen in Figs. 4(b) and 4(d), where fewer green structures are visible. The CoGEN’s ability to generate these compensatory structures suggests that it maintains a more realistic distribution of SGS vortical structures in the reconstructed flow field, even when working with more heavily filtered inputs. This improved performance stems from the CoGEN’s awareness of the k_{SGS} .

In Secs. III B and III C, we will examine the statistical equivalence between the generated data and the ground truth DNS data to assess the physical accuracy of these generated turbulent structures.

B. Turbulent flow statistics

To quantitatively examine the statistical properties of coherent structures in the reconstructed turbulent flow, we analyze the probability density functions (PDFs) of the Q-criterion. Figure 5 shows the PDFs of Q normalized by the square of the Kolmogorov time scale τ_η^2 for various cases. Figure 5(a) compares the PDFs for the DNS data with those of the reconstructed fields using CoGEN and TradGEN networks at filtering levels $\sigma = 4\Delta$. Figure 5(b) presents the same for filtering levels $\sigma = 2\Delta$. Additionally, the figure includes the PDFs for the FDNS fields, FDNS-2 and FDNS-4, for reference.

It is evident from the figure that the filtering process attenuates extreme events characteristic of turbulent intermittency, where intense rotation (positive Q) and strain (negative Q) occur. Consequently, the distribution narrows for both positive and negative Q values, with FDNS-4 exhibiting a more pronounced effect than FDNS-2, consistent with its stronger filtering.

Examining the performance of TradGEN, we observe that TradGEN-2 exhibits remarkable fidelity, closely replicating the DNS distribution across the range of Q values shown in the diagram. This suggests that for moderate filtering ($\sigma = 2\Delta$), the TradGEN effectively recovers the statistical properties of coherent structures, including intermittent events in both rotation-dominated and strain-dominated regions. In contrast, TradGEN-4 only partially reconstructs the intermittent events, significantly underestimating the probability of extreme Q values in the tails of the PDF. This reveals the TradGEN’s limitation in recovering structures associated with intense rotational and strain events from heavily filtered inputs ($\sigma = 4\Delta$).

The results of CoGEN reveal that both CoGEN-2 and CoGEN-4 closely match the DNS distribution across the range of Q values, except for a marginal overprediction by CoGEN-2 in the negative Q region. The CoGEN effectively recovers the statistical properties of coherent structures, including intense intermittent events, with reduced sensitivity to the filtering level. This suggests that the conditional information (k_{SGS}) enables robust recovery of fine-scale structures and extreme events, even from heavily filtered inputs ($\sigma = 4\Delta$). Overall, these results demonstrate the superior performance of the CoGEN over TradGEN in reconstructing the statistics of coherent structures from heavily filtered turbulent flow data.

Turbulence kinetic energy spectra provide a quantitative measure of how kinetic energy is distributed across different spatial scales in turbulent flows. Figure 6 shows the turbulence kinetic energy spectra of the reconstructed velocity fields with that of the ground truth DNS velocity field. The reconstruction performance of two types of generators, the CoGEN and the TradGEN, is compared. Additionally, the figure includes the energy spectra of the FDNS velocity fields, FDNS-2 and FDNS-4, for reference.

As expected, the FDNS energy spectra deviate from the DNS spectra due to the filtering process, with FDNS-4 exhibiting a more pronounced deviation than FDNS-2, especially at high wavenumbers. It is seen from Fig. 6 that the energy spectra from both TradGEN and CoGEN models, using different FDNS inputs, closely match the DNS results in low wave numbers. This observation is consistent with the vortex structure visualization in Fig. 4, where both TradGEN and CoGEN successfully reconstruct the large-scale structures, which are associated with low wavenumbers. However, at high wavenumbers, the TradGEN significantly under-predicts the turbulence kinetic energy compared to the DNS for both FDNS inputs. This under-prediction becomes more pronounced, both in magnitude and range,

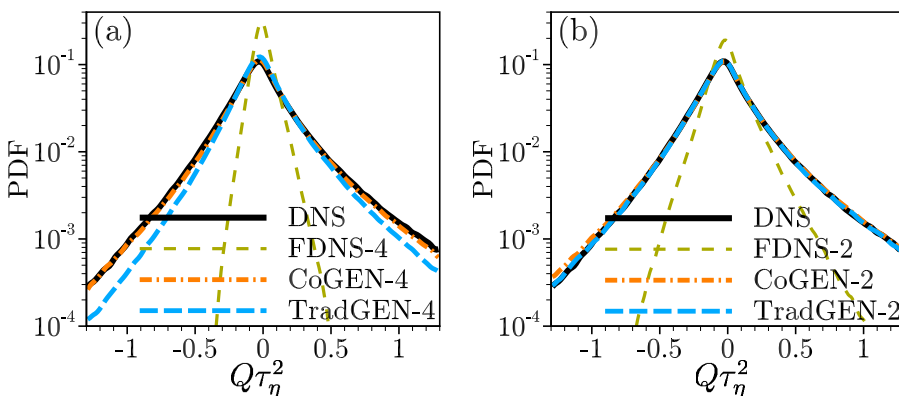


FIG. 5. PDF of the normalized second invariant of the velocity gradient tensor ($Q\tau_\eta^2$). (a) Comparison of DNS with FDNS-4, CoGEN-4, and TradGEN-4 reconstructions ($\sigma = 4\Delta$). (b) Comparison of DNS with FDNS-2, CoGEN-2, and TradGEN-2 reconstructions ($\sigma = 2\Delta$).

17 December 2024 11:15:52

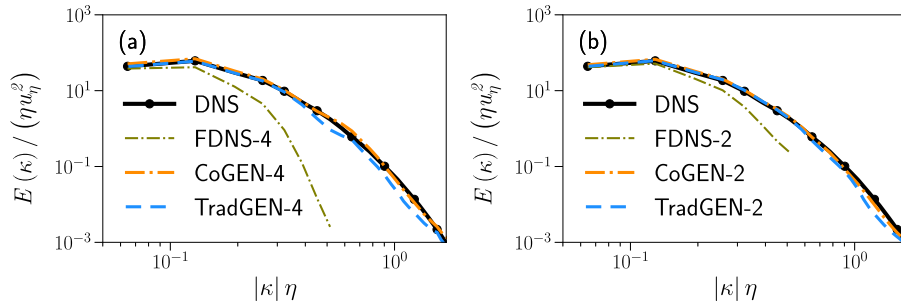


FIG. 6. Turbulent energy spectra of reconstructed velocity fields compared to DNS. (a) Results using FDNS-4 input. (b) Results using FDNS-2 input.

for TradGEN-4. The reason for this behavior is that as σ increases, the FDNS velocity field becomes smoother, losing more grid-scale information near the filter cutoff. Consequently, the TradGEN fails to recover many small-scale structures, as evident from the increased number of omitted structures (blue vortices) in Fig. 4(d) compared to Fig. 4(e). In contrast, the CoGEN, being aware of k_{sgs} , provides a more accurate prediction of the energy spectra at high wavenumbers for both levels of smoothness in FDNS input. When σ increases, k_{sgs} increases as well, allowing the CoGEN to accurately determine how much energy should be recovered. This enables the CoGEN to generate additional compensating structures [green vortices in Fig. 4(e)] to maintain the overall energy balance.

To assess how well our reconstructed fields capture the multi-scale characteristics of turbulence, we calculate the PDF of the dissipation multiplier.⁵⁴ The dissipation multiplier, defined as $M = \epsilon(\Delta)/\epsilon(2\Delta)$, represents the ratio of dissipation (ϵ) in boxes of size Δ^3 and $(2\Delta)^3$. A PDF that closely matches that of the DNS indicates that the reconstructed field accurately reproduces the intermittent spatial distribution of dissipation.

Figure 7 shows the PDFs of the dissipation multiplier M for DNS data and reconstructed fields using conditional and traditional generators at filtering levels $\sigma = 4\Delta$ (a) and $\sigma = 2\Delta$ (b). The distributions center around $M \approx 1/8$, reflecting the geometric division in 3D turbulence where each $(2\Delta)^3$ box splits into $8(\Delta)^3$ boxes. In the case of a uniform ϵ distribution, the PDF would be just a delta function at $M = 1/8$.

Since the total integral of the PDF distribution is equal to one, overestimating the PDF peak at $M = 1/8$ means underestimating the PDF at small and large M , i.e., the underestimation of the dissipation rate intermittence. Insets reveal that both CoGEN and TradGEN overestimate peak probabilities compared to DNS, with TradGEN showing

more pronounced overestimation, especially for $\sigma = 4\Delta$. The TradGEN distribution is narrower with a higher, sharper peak, suggesting it underestimates the variability in dissipation ratios between scales and, thus, the intensity of intermittent events. CoGEN's closer match to DNS demonstrates that conditional information (k_{sgs}) improves reconstruction of multi-scale dissipation patterns. This aligns with our earlier findings on energy spectra and small-scale structures.

C. Dispersed-phase statistics

In the present section, we assess the fidelity of our reconstructed velocity fields for turbulent particle dispersion. Particle dispersion stems from the random nature of small-scale turbulent eddies. Thus, a pair of particles that are initially close to each other can soon experience different velocity fluctuations, which lead to divergent trajectory paths. By comparing the statistics of particle pair dispersion, one can assess the extent to which the reconstructed fields are reliable for particle tracking.

Assuming drag force dominance, the governing equations for each particle's position and velocity are

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{u}_p, \quad (13)$$

$$\frac{d\mathbf{u}_p}{dt} = \frac{\mathbf{u}_r}{\tau_p/f_d}. \quad (14)$$

Here, $\mathbf{u}_r = \mathbf{u}_s - \mathbf{u}_p$ denotes the relative velocity, where \mathbf{u}_s is the fluid velocity experienced by the particle, obtained through interpolation of the carrier velocity field at the particle's location. The term $\tau_p = \rho_p d_p^2 / 18\mu$ represents the particle relaxation time scale, and the

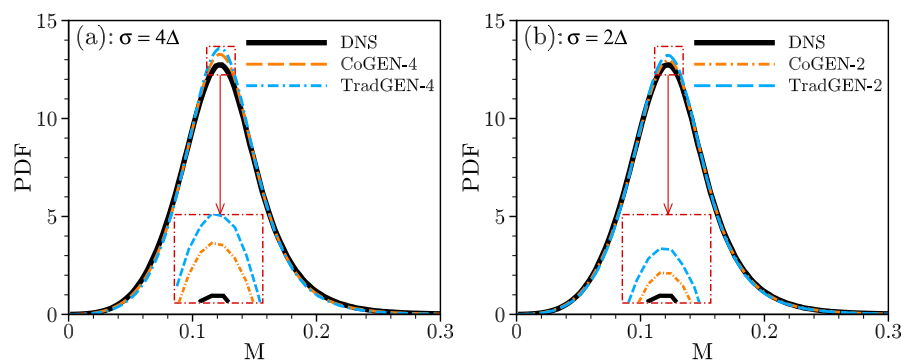


FIG. 7. PDFs of dissipation multiplier M for DNS and reconstructed fields using CoGEN and TradGEN. (a) $\sigma = 4\Delta$ and (b) $\sigma = 2\Delta$. Insets show magnified views of peak regions, highlighting differences in distribution shapes among DNS, CoGEN, and TradGEN models.

drag coefficient, f_d , is calculated using the Schiller and Naumann correlation⁵⁵ for spherical particles:

$$f_d = \begin{cases} 1 & \text{for } Re_p \leq 1, \\ 1 + 0.15Re_p^{0.687} & \text{for } 1 < Re_p \leq 1000, \\ \frac{0.44Re_p}{24} & \text{for } Re_p > 1000, \end{cases} \quad (15)$$

where $Re_p = |\mathbf{u}_r|d_p/\nu$ is the particle Reynolds number.

To calculate the expected dispersion of particle pairs, we initially inject 4096 pairs with a separation distance of 0.5η . These particle pairs are randomly distributed throughout the domain and tracked in the carrier velocity field using the Lagrangian solver of OpenFOAM. We investigate three different particle inertia levels, characterized by Stokes numbers $St_k = 0.125, 1$, and 8 . The corresponding particle diameters, non-dimensionalized by η , are $d_p/\eta = 0.029, 0.081$, and 0.23 , respectively. The Stokes number, defined as $St_k = \tau_p/\tau_\eta$, represents the ratio of the particle relaxation time to the characteristic turbulent time scale of the flow, which in this case is the Kolmogorov time scale. During the simulation, the ensemble averaged distance between paired particles, denoted as $\langle \delta \rangle$, is calculated for each time step. This particle tracking process is performed for different carrier velocity fields, including the DNS, FDNS, TradGEN, and CoGEN velocity fields with the both levels of filtering ($\sigma = 2\Delta$ and 4Δ).

Figure 8 shows the temporal evolution of particle pair dispersion, represented by $\langle \delta \rangle/\eta$ over t/τ_η . The FDNS significantly underpredicts particle dispersion across all cases, with the effect more pronounced for the stronger filter ($\sigma = 4\Delta$) as expected. For the moderate filtering ($\sigma = 2\Delta$), both the conditional and traditional generators perform

remarkably well, closely matching the DNS dispersion curves for all Stokes numbers. This aligns with our previous observations of accurate reconstruction of vortical structures for this filter size. The success in reproducing particle dispersion further validates the physical fidelity of the reconstructed fields, as particle motion integrates the effects of multi-scale turbulent structures over time. The advantages of the conditional generator become apparent only with the stronger filter ($\sigma = 4\Delta$). In this case, the CoGEN maintains excellent agreement with DNS across all Stokes numbers, while the TradGEN shows noticeable deviations for smaller Stokes numbers. This performance difference aligns with our vortex reconstruction results (Fig. 4), where CoGEN more effectively recovered fine-scale structures for $\sigma = 4\Delta$. The CoGEN's superior performance stems from its use of SGS kinetic energy information, enabling more accurate reconstruction of small-scale fluctuations critical for particle dispersion with lower Stokes numbers.

IV. CONCLUSION

This study introduces CESRGAN—a conditional deep learning model—for reconstructing high-resolution turbulent velocity fields from low-resolution inputs. CESRGAN consists of a conditional discriminator and a conditional generator, the latter being called CoGEN.

By incorporating subgrid-scale turbulence kinetic energy as a condition for CoGEN, our approach significantly improves the recovery of small-scale turbulence structures compared to traditional super-resolution methods. Being aware of k_{SGS} , CoGEN reconstructs the fine-scale structures according to the desired turbulence intensity, enabling the model to upscale inputs with different degrees of smoothness to achieve the target turbulence kinetic energy level.

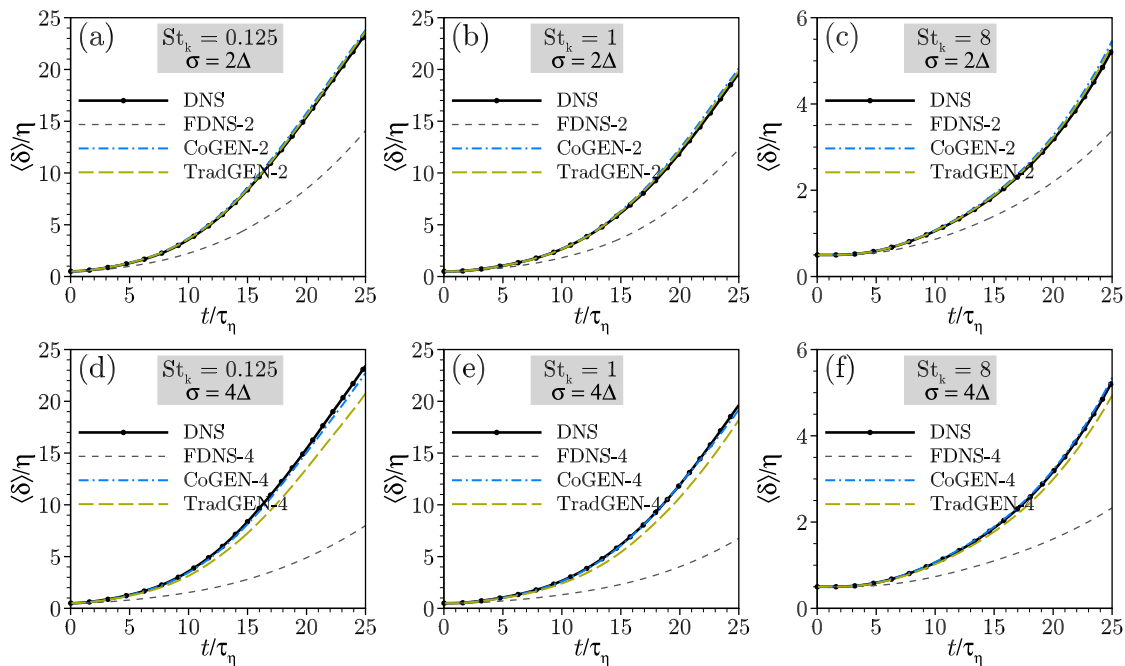


FIG. 8. Evolution of particle pair dispersion. Panels show $\langle \delta \rangle/\eta$ against t/τ_η for $St_k = 0.125$ (a) and (d), 1 (b) and (e), and 8 (c) and (f), with filter widths $\sigma = 2\Delta$ (top) and 4Δ (bottom). Pairs initially separated by 0.5η .

The analysis of Q-criterion isosurfaces showed that the present CoGEN more effectively reconstructed fine-scale vortical structures than the traditional TradGEN, particularly for heavily filtered inputs. This improved small-scale reconstruction was reflected in the energy spectra, where CoGEN-reconstructed fields showed better agreement with DNS at high wavenumbers.

The CoGEN also more accurately captured the intermittent nature of turbulent flows, as evidenced by the Q-criterion PDFs. These PDFs revealed better reproduction of extreme events in both rotation-dominated and strain-dominated regions across different filtering levels. Similarly, the dissipation multiplier PDFs demonstrated CoGEN's superior ability to reproduce the multi-scale characteristics of turbulence.

Particle dispersion simulations further validated the physical fidelity of the CoGEN-reconstructed fields. The results from CoGEN closely matched DNS across various Stokes numbers and filtering levels, highlighting the potential application of super-resolution models in particle-laden flows.

Our findings emphasize the value of incorporating physical information, such as k_{SGS} , into super-resolution models for turbulent flows. This approach not only improves reconstruction accuracy but also enhances the model's robustness in dealing with different levels of input filtering. This opens the way for reconstructing fine-scale turbulence from LES, which can have various resolutions and simultaneously allows for obtaining k_{SGS} .

While CESRGAN demonstrates promising results, it is constrained by its reliance on structured, uniform grids—a limitation inherent to CNN-based methods. This restricts its applicability in complex geometries. Future research directions include adapting the model for unstructured grids, potentially through the use of graph convolutional networks. Such advancements could significantly expand CESRGAN's utility in a broader range of turbulent flow simulations and practical engineering applications.

ACKNOWLEDGMENTS

This work was performed on the HoreKa supercomputer funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research. The authors acknowledge support by the state of Baden-Württemberg through bwHPC.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Hesam Tofighian: Conceptualization (lead); Formal analysis (lead); Investigation (lead); Methodology (lead); Software (lead); Visualization (lead); Writing – original draft (lead). **Jordan A. Denev:** Resources (equal); Supervision (equal); Writing – review & editing (equal). **Nikolai Kornev:** Project administration (lead); Resources (equal); Supervision (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- C. Marchioli, M. V. Salvetti, and A. Soldati, "Some issues concerning large-eddy simulation of inertial particle dispersion in turbulent bounded flows," *Phys. Fluids* **20**(4), 040603 (2008).
- J. G. M. Kuerten, "Point-particle DNS and LES of particle-laden turbulent flow—a state-of-the-art review," *Flow, Turbul. Combust.* **97**, 689–713 (2016).
- Q. Wang and K. D. Squires, "Large eddy simulation of particle-laden turbulent channel flow," *Phys. Fluids* **8**(5), 1207–1223 (1996).
- J.-P. Minier, "On Lagrangian stochastic methods for turbulent polydisperse two-phase reactive flows," *Prog. Energy Combust. Sci.* **50**, 1–62 (2015).
- C. Marchioli, "Large-eddy simulation of turbulent dispersed flows: A review of modelling approaches," *Acta Mech.* **228**(3), 741–771 (2017).
- H. Tofighian, E. Amani, and M. Saffar-Avval, "A large eddy simulation study of cyclones: The effect of sub-models on efficiency and erosion prediction," *Powder Technol.* **360**, 1237–1252 (2020).
- S. Stolz, N. A. Adams, and L. Kleiser, "An approximate deconvolution model for large-eddy simulation with application to incompressible wall-bounded flows," *Phys. Fluids* **13**(4), 997–1015 (2001).
- J. G. M. Kuerten, "Subgrid modeling in particle-laden channel flow," *Phys. Fluids* **18**(2), 025108 (2006).
- M. Bassenne, M. Esmaily, D. Livescu, P. Moin, and J. Urzay, "A dynamic spectrally enriched subgrid-scale model for preferential concentration in particle-laden turbulence," *Int. J. Multiphase Flow* **116**, 270–280 (2019).
- A. Scotti and C. Meneveau, "A fractal model for large eddy simulation of turbulent flow," *Physica D* **127**(3–4), 198–232 (1999).
- E. O. Akinlabi, M. Waclawczyk, S. P. Malinowski, and J. P. Mellado, "Fractal reconstruction of sub-grid scales for large eddy simulation," *Flow, Turbul. Combust.* **103**, 293–322 (2019).
- P. Flohr and J. C. Vassilicos, "A scalar subgrid model with flow structure for large-eddy simulations of scalar variances," *J. Fluid Mech.* **407**, 315–349 (2000).
- B. Ray and L. R. Collins, "A subgrid model for clustering of high-inertia particles in large-eddy simulations of turbulence," *J. Turbul.* **15**(6), 366–385 (2014).
- A. S. Ghate and S. K. Lele, "Subfilter-scale enrichment of planetary boundary layer large eddy simulation using discrete Fourier–Gabor modes," *J. Fluid Mech.* **819**, 494–539 (2017).
- A. S. Ghate and S. Lele, "Gabor mode enrichment in large eddy simulations of turbulent flow," *J. Fluid Mech.* **903**, A13 (2020).
- M. Hausmann, F. Evrard, and B. Wachem, "Wavelet-based modeling of subgrid scales in large-eddy simulation of particle-laden turbulent flows," *Phys. Rev. Fluids* **8**(10), 104604 (2023).
- C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(2), 295–307 (2016).
- C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Computer Vision—ECCV 2016: 14th European Conference*, October 11–14, 2016, *Proceedings, Part II* (Springer, Amsterdam, The Netherlands, 2016), Vol. 14, pp. 391–407.
- J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2016), pp. 1646–1654.
- W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2016), pp. 1874–1883.
- B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (IEEE, 2017), pp. 136–144.
- Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2017), pp. 3147–3155.
- C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2017), pp. 4681–4690.

- ²⁴I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM* **63**(11), 139–144 (2020).
- ²⁵K. Singla, R. Pandey, and U. Ghanekar, "A review on Single Image Super Resolution techniques using generative adversarial network," *Optik* **266**, 169607 (2022).
- ²⁶X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "ESRGAN: Enhanced super-resolution generative adversarial networks," in Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018.
- ²⁷A. Jolicœur-Martineau, "The relativistic discriminator: A key element missing from standard GAN," [arXiv:1807.00734](https://arxiv.org/abs/1807.00734) (2018).
- ²⁸K. Fukami, K. Fukagata, and K. Taira, "Super-resolution reconstruction of turbulent flows with machine learning," *J. Fluid Mech.* **870**, 106–120 (2019).
- ²⁹K. Fukami, K. Fukagata, and K. Taira, "Assessment of supervised machine learning methods for fluid flows," *Theor. Comput. Fluid Dyn.* **34**(4), 497–519 (2020).
- ³⁰B. Liu, J. Tang, H. Huang, and X.-Y. Lu, "Deep learning methods for super-resolution reconstruction of turbulent flows," *Phys. Fluids* **32**(2), 025105 (2020).
- ³¹Z. Zhou, B. Li, X. Yang, and Z. Yang, "A robust super-resolution reconstruction model of turbulent flow data based on deep learning," *Comput. Fluids* **239**, 105382 (2022).
- ³²A. Subramaniam, M. L. Wong, R. D. Borker, S. Nimmagadda, and S. K. Lele, "Turbulence enrichment using physics-informed generative adversarial networks," [arXiv:2003.01907](https://arxiv.org/abs/2003.01907) (2020).
- ³³H. Kim, J. Kim, S. Won, and C. Lee, "Unsupervised deep learning for super-resolution reconstruction of turbulence," *J. Fluid Mech.* **910**, A29 (2021).
- ³⁴M. Bode, M. Gauding, K. Kleinheinz, and H. Pitsch, "Deep learning at scale for sub-grid modeling in turbulent flows: Regression and reconstruction," in *International Conference on High Performance Computing* (Springer, 2019), pp. 541–560.
- ³⁵Z. Deng, C. He, Y. Liu, and K. C. Kim, "Super-resolution reconstruction of turbulent velocity fields using a generative adversarial network-based artificial intelligence framework," *Phys. Fluids* **31**(12), 125111 (2019).
- ³⁶M. Bode, M. Gauding, Z. Lian, D. Denker, M. Davidovic, K. Kleinheinz, J. Jitsev, and H. Pitsch, "Using physics-informed enhanced super-resolution generative adversarial networks for subfilter modeling in turbulent reactive flows," *Proc. Combust. Inst.* **38**(2), 2617–2625 (2021).
- ³⁷L. Yu, M. Z. Yousif, M. Zhang, S. Hoyas, R. Vinuesa, and H.-C. Lim, "Three-dimensional ESRGAN for super-resolution reconstruction of turbulent flows with tricubic interpolation-based transfer learning," *Phys. Fluids* **34**(12), 125126 (2022).
- ³⁸L. Nista, C. D. K. Schumann, T. Grenga, A. Attili, and H. Pitsch, "Investigation of the generalization capability of a generative adversarial network for large eddy simulation of turbulent premixed reacting flows," *Proc. Combust. Inst.* **39**(4), 5279–5288 (2023).
- ³⁹M. Bode, M. Gauding, D. Goeb, T. Falkenstein, and H. Pitsch, "Applying physics-informed enhanced super-resolution generative adversarial networks to turbulent premixed combustion and engine-like flame kernel direct numerical simulation data," *Proc. Combust. Inst.* **39**(4), 5289–5298 (2023).
- ⁴⁰L. Nista, H. Pitsch, C. D. K. Schumann, M. Bode, T. Grenga, J. F. MacArt, and A. Attili, "Influence of adversarial training on super-resolution turbulence reconstruction," *Phys. Rev. Fluids* **9**(6), 064601 (2024).
- ⁴¹M. Bode and J. H. Göbbert, "Acceleration of complex high-performance computing ensemble simulations with super-resolution-based subfilter models," *Comput. Fluids* **271**, 106150 (2024).
- ⁴²M. Bode, "AI super-resolution subfilter modeling for multi-physics flows," in Proceedings of the Platform for Advanced Scientific Computing Conference, 2023.
- ⁴³M. Z. Yousif, L. Yu, and H.-C. Lim, "High-fidelity reconstruction of turbulent flow from spatially limited data using enhanced super-resolution generative adversarial network," *Phys. Fluids* **33**(12), 125119 (2021).
- ⁴⁴T. L. Trinh, F. Chen, T. Nanri, and K. Akasaka, "3D super-resolution model for vehicle flow field enrichment," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (IEEE, 2024), pp. 5826–5835.
- ⁴⁵K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2016), pp. 770–778.
- ⁴⁶X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Advance in Neural Information Processing Systems* (Neural Information Processing Systems Foundation, Inc., 2016), Vol. 29.
- ⁴⁷W. Yang, X. Zhang, Y. Tian, W. Wang, J.-H. Xue, and Q. Liao, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia* **21**(12), 3106–3121 (2019).
- ⁴⁸Y. Jo, S. W. Oh, P. Vajda, and S. J. Kim, "Tackling the ill-posedness of super-resolution through adaptive target generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2021), pp. 16236–16245.
- ⁴⁹X. Zhang, H. Song, K. Zhang, J. Qiao, and Q. Liu, "Single image super-resolution with enhanced Laplacian pyramid network via conditional generative adversarial learning," *Neurocomputing* **398**, 531–538 (2020).
- ⁵⁰H. Yin, "Comparison and analysis of various deep learning models for super-resolution reconstruction of turbulent flows," *J. Phys.* **2634**, 012046 (2023).
- ⁵¹V. Eswaran and S. B. Pope, "An examination of forcing in direct numerical simulations of turbulence," *Comput. Fluids* **16**(3), 257–278 (1988).
- ⁵²C. Fureby, G. Tabor, H. G. Weller, and A. D. Gosman, "A comparative study of subgrid scale models in homogeneous isotropic turbulence," *Phys. Fluids* **9**(5), 1416–1429 (1997).
- ⁵³S. Park and E. Lee, "One-to-many approach for improving super-resolution," [arXiv:2106.10437](https://arxiv.org/abs/2106.10437) (2021).
- ⁵⁴N. Kornev, V. Zhdanov, and E. Hassel, "Study of scalar macro- and microstructures in a confined jet," *Int. J. Heat Fluid Flow* **29**(3), 665–674 (2008).
- ⁵⁵L. Schiller and A. Naumann, "A drag coefficient correlation," *Z. Ver. Dtsch. Ing.* **77**, 318–320 (1935).