Computers and Fluids 239 (2022) 105382

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

A robust super-resolution reconstruction model of turbulent flow data based on deep learning

Zhideng Zhou, Binglin Li, Xiaolei Yang, Zixuan Yang *

The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
School of Engineering Sciences, University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Keywords:
Super-resolution model
Direct numerical simulation
Large-Eddy simulation
Isotropic turbulence
Unresolved scales

ABSTRACT

A new super-resolution model, namely the turbulence volumetric super-resolution (TVSR) model, is developed based on convolutional neural network (CNN) to reconstruct three-dimensional high-resolution turbulent flow field data from low-resolution data. Direct numerical simulation (DNS) and corresponding filtered DNS (FDNS) data of homogeneous isotropic turbulence at various Reynolds numbers are used to train the TVSR model. The proposed model is a modification of Liu et al. (2020), aiming to provide an improved generalization capability of the super-resolution model. For this purpose, we propose a patchwise training strategy in consideration of the property of turbulence that the velocity correlation between two points diminishes as the separation becomes sufficiently large. Furthermore, data at various Reynolds numbers are combined together to train the model. In comparison with existing models, the present TVSR model shows a better generalization capability in two aspects. First, the TVSR model trained using data at low Reynolds numbers is found robust and accurate in the super-resolution reconstructions of flow fields at higher Reynolds numbers. Second, although only DNS data are used for training, the TVSR model is also robust in reconstructing high-resolution flow fields from low-resolution data obtained from large-eddy simulation (LES). This feature of the TVSR model provides a new access to obtain turbulent motions at unresolved scales in LES studies of turbulent flows.

1. Introduction

High-resolution flow data are valuable for investigations of flow structures and statistics of turbulent flows. In the context of computational fluid dynamics, the direct numerical simulation (DNS) resolves turbulent motions at all scales and provides high-resolution flow data. However, it is expensive to apply DNS in practical flows at high Reynolds numbers, because the computational cost of DNS increases rapidly at a rate of the cube of the Reynolds number [1–3]. As a compromise, the large-eddy simulation (LES) resolves turbulent motions at large scales using grids with relatively low resolution to reduce the computational cost, while the effects of unresolved motions on resolved flow fields are approximated using a subgrid-scale model [4,5]. However, in certain applications, such as turbulent acoustics and transport of small particles [6–10], the small-scale unresolved motions can be also critical. In the present study, we aim at reconstructing turbulent motions at all scales from low-resolution flow fields obtained from LES.

There are various flow field reconstruction models, including the approximate deconvolution method (ADM), stochastic model, kinematic simulation (KS) model, hybrid models and Gabor mode enrichment model. In the ADM [11,12], the unresolved flow field is reconstructed as a truncated expansion of the inverse filter operator with respect to the filter operator, which improves the LES result of the flow field at resolved scales. However, as noted by Cernick et al. [13], the ADM does not reconstruct turbulent motions at unresolved scales. In stochastic models [14,15], the subgrid-scale flow field is treated as a white noise given by the solution of the Langevin equation that governs the Brownian motion supplemented with random sources [16]. Recently, Barge and Gorokhovski [17] proposed a stochastic subgrid acceleration model by employing the LES equations with a forcing term on the smallest resolved scales to simulate the effects of the dynamics at subgrid scales. In KS models [18–22], the turbulent-like flows are generated using a series of random Fourier modes, of which the energy spectra follow a prescribed $-5/3$-scaling law at the inertial subrange. The hybrid ADM/KS model [23] reconstructs the energy spectra at both resolved and subgrid scales and accurately predicts the Lagrangian statistical properties, but it only generates Gaussian turbulence without recovering the small-scale intermittency of turbulence [24]. Moreover, Ghate and Lele [25] developed the turbulence enrichment approach using the spectrally localized Gabor modes, which offers an optimal basis to represent small-scale turbulence within quasi-homogeneous regions.

* Corresponding author at: The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China.
E-mail address: yangzx@imech.ac.cn (Z. Yang).

https://doi.org/10.1016/j.compfluid.2022.105382
Received 16 June 2021; Received in revised form 23 November 2021; Accepted 25 February 2022
Available online 9 March 2022
0045-7930/© 2022 Elsevier Ltd. All rights reserved.

This approach showed promising results in the LES of two problems at high Reynolds numbers, namely the homogeneous isotropic turbulence and a rough-wall turbulent boundary layer flow [26].

In recent years, machine learning is applied to solve various problems in turbulence research [27,28], including the development of turbulence models [29–31], temporal prediction of turbulence [32,33], turbulence identification [34,35], turbulent flow control [36,37], and flow field reconstruction [38–47]. In the following content, we focus on the applications of machine learning to the flow field reconstruction.

Maulik and San [38] proposed to use a single-layer feed-forward artificial neural network (ANN) to recover an original flow field from artificially prescribed perturbations (in terms of applying a Gaussian filter or adding white noise). Their model performed well in the *a priori* tests of two-dimensional (2D) homogeneous isotropic turbulence, three-dimensional (3D) decaying homogeneous isotropic turbulence, and compressible stratified turbulence. The ANN was then used as convolutional and deconvolutional maps between a subsampled DNS flow field and a filtered DNS flow field to close the 2D LES equations [39]. Beck et al. [40] used the ANN and a convolutional neural network (CNN) to learn the filtered momentum fluxes in LES of decaying homogeneous isotropic turbulence. They found that the CNN performed well in reconstructing the closure terms. Xie et al. [48] proposed a spatially multi-scale ANN to reconstruct the SGS stress and heat flux of compressible isotropic turbulence, which showed an advantage over the dynamic mixed model in the prediction of the spectra of velocity and temperature. In the above applications of ANN and CNN to the flow field reconstruction, the resolution of the reconstructed flow variable is limited to the same as that of the input data, while the turbulent motions at unresolved scales corresponding to a higher resolution remain unavailable.

Fukami et al. [41] conducted super-resolution reconstruction of turbulent flow field using two different machine learning approaches, namely the CNN and the hybrid downsampled skip-connection/multi-scale models. These models were examined in the context of 2D cylinder wake flow and 2D decaying isotropic turbulence, which showed the capability of machine learning in reconstructing high-resolution laminar and turbulent flow fields from low-resolution data. Deng et al. [42] applied the enhanced super-resolution generative adversarial network (ESRGAN) to reconstruct the flow around a single cylinder and the wake flow of two side-by-side cylinders obtained from laboratory experiments. The analyses of instantaneous fields, statistical flow quantities, and spatial correlations showed that high-resolution flow fields were reasonably reconstructed. The ESRGAN was also applied to reconstruct the 3D velocity fields of homogeneous isotropic turbulence [44] and decaying turbulence [45]. In these applications, the reconstructed flow field showed visually realistic image. Recently, Liu et al. [47] used static CNN (SCNN) and multiple temporal paths CNN (MTPCNN) models for the super-resolution reconstruction in 3D homogeneous isotropic turbulence. The energy spectra and PDF of the velocity gradients obtained from the MTPCNN models were closer to the DNS results than those obtained form SCNN. Kim et al. [49] used a cycle-consistent generative adversarial network (CycleGAN) to train an unsupervised super-resolution model, which showed an excellent ability on reconstructing small-scale structures of isotropic turbulence and channel turbulence. The models of Liu et al. [47] and Kim et al. [49] were both essentially 2D models, while the 3D high-resolution flow field is reconstructed by applying the model slice by slice to 2D low-resolution flow fields.

Inspired by the above pioneering works on the super-resolution models, we propose a turbulence volumetric super-resolution (TVSR) model using CNN to reconstruct the 3D high-resolution homogeneous isotropic turbulence from the low-resolution flow data by modifying the super-resolution model of Liu et al. [47]. In comparison with the existing super-resolution models, the proposed model is different mainly in the following two aspects. First, the proposed model applies a patchwise training strategy, considering that the velocity correlation between two points diminishes as the separation becomes sufficiently large. This strategy also enables the training of TVSR model using data at different Reynolds numbers together, and thus provides a possibility of applying the model to higher Reynolds numbers. Second, the proposed model directly processes 3D flow fields, instead of dealing with 2D slices as in previous studies. This treatment is physically more reasonable, considering that the turbulence is a three-dimensional flow phenomenon. Meanwhile, for the same grid ratio r, the available information of low-resolution data is $1/r^2$ and $1/r^3$ that of high-resolution data for 2D and 3D fields, respectively. As a result, it is computationally more expensive to train a 3D model than a 2D model. The performance of the TVSR model is evaluated by comparing the instantaneous vortex structures and turbulent statistics obtained from the TVSR model and DNS. In the *a priori* tests, the reconstructions of filtered DNS (FDNS) snapshots at various Reynolds numbers are performed to examine the accuracy of the model. In the *a posteriori* test, the TVSR model is applied to the super-resolution reconstructions of flow fields obtained from LES. The remainder of this paper is organized as follows: In Section 2, the numerical details of the DNS and LES of isotropic turbulence and the data for training and validating the models are described. The training approach and parameters of the TVSR model are given in Section 3. The *a priori* and *a posteriori* test results are presented in Section 4, with the conclusions of this research given in Section 5.

2. Data preparation

In this section, we describe the data of homogeneous isotropic turbulence used for training and validating the TVSR model. The governing equations and numerical methods employed for conducting DNS and LES are given in Sections 2.1 and 2.2, respectively. The filters used for generating low-resolution FDNS data are introduced in Section 2.3.

2.1. Direct numerical simulation

The DNS data are generated by solving the following Navier–Stokes equations for incompressible flows

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{u} \times \boldsymbol{\omega} - \nabla \left(\frac{p}{\rho} + \frac{1}{2}\mathbf{u}^2 \right) + \nu \nabla^2 \mathbf{u} + \mathbf{f}(\mathbf{x}, t), \tag{2}$$

where $\mathbf{u}$ denotes the velocity, $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ is the vorticity, p is the pressure, and ρ and ν represent the density and kinematic viscosity of fluid, respectively. The turbulent flow is sustained by a deterministic forcing term $\mathbf{f}(\mathbf{x}, t)$, of which the value is nonzero at Fourier modes with wavenumber magnitudes no larger than 2.0. With this deterministic forcing, statistically steady turbulence is generated by maintaining constant energy production inside the first two wavenumber shells, and the energy ratio between the two shells follows the $k^{-5/3}$ scaling law of the energy spectra [50].

The DNS of homogeneous isotropic turbulence is conducted using a standard pseudo-spectral method code in a periodic cubic domain, of which the length of each edge is L. The computational domain is discretized uniformly using N^3 grid points. In Fourier space, Eqs. (1)~(2) are expressed as

$$\left(\frac{\partial}{\partial t} + \nu k^2 \right) \hat{\mathbf{u}}(\mathbf{k}, t) = \mathbf{P}(\mathbf{k}) \mathcal{F}(\mathbf{u} \times \boldsymbol{\omega}) + \hat{\mathbf{f}}(\mathbf{k}, t), \tag{3}$$

where $\hat{\mathbf{u}}(\mathbf{k}, t) = \mathcal{F}(\mathbf{u})$ denotes the Fourier coefficients of the velocity, with $\mathcal{F}$ being the Fourier transformation operator. The projection tensor $\mathbf{P}(\mathbf{k}) = \delta_{ij} - k_i k_j / k^2 \, (i, j = 1, 2, 3)$ projects $\mathcal{F}(\mathbf{u} \times \boldsymbol{\omega})$ onto the plane normal to the wavenumber vector $\mathbf{k} = [k_1, k_2, k_3]$ and eliminates the pressure gradient term in Eq. (2). The wavenumber components in Fourier space are defined as $k_j = n_j (2\pi/L)$, where $n_j = -N/2, \ldots, -1, 0, 1, \ldots, N/2 - 1$ for $j = 1, 2, 3$. In physical space, the grid is evenly spaced in all three directions, and its resolution is $\Delta = L/N$. Due to the use of the 2/3 rule for eliminating the aliasing

Table 1
DNS parameters of isotropic turbulent flows in a statistically steady state.

Re_λ	N^3	Δ/L
64.43	128^3	0.0491
128.78	256^3	0.0245
205.51	512^3	0.0123
302.04	1024^3	0.00614

error, the maximum resolvable wavenumber is $k_{\max} = 2\pi/(3\Delta)$ [23,51]. The spatial resolution is monitored by the value of $k_{\max}\eta$, where η is the Kolmogorov length scale. The value of $k_{\max}\eta$ should be larger than 1.0 for small-scale turbulent motions being well resolved, and $k_{\max}\eta = 1.3$ is utilized in all DNS cases in the present study. The fluid velocity in Fourier space is advanced in time using a second-order Adams–Bashforth method and an exact integration for the nonlinear terms and the linear viscous term, respectively. The time step is chosen to ensure that the Courant–Friedrichs–Lewy (CFL) number is 0.5 or less for numerical stability.

Table 1 summarizes the parameters of DNS cases. As shown, we have conducted four DNS cases. The Reynolds number $Re_\lambda = u'\lambda/\nu$ ranges from 64.43 to 302.04, where $u' = \sqrt{\langle u_i u_i \rangle/3}$ is the velocity fluctuation intensity, and $\lambda = \left(15\nu u'^2/\varepsilon\right)^{1/2}$ is the Taylor microscale, with ε being the energy dissipation rate. The number of grid points is 128, 256, 512, 1024, respectively, which yields a finer grid resolution Δ/L for cases at higher Reynolds numbers.

2.2. Large eddy simulation

The LES of homogeneous isotropic turbulence is performed using the same code for DNS with lower grid resolution. The governing equation in Fourier space for the LES is expressed as

$$\left(\frac{\partial}{\partial t} + \left[\nu + \nu_e\left(k\,|k_c\right)\right]k^2\right)\hat{\bar{\mathbf{u}}}(\mathbf{k},t) = \mathbf{P}(\mathbf{k})\,\mathcal{F}\left(\bar{\mathbf{u}}\times\bar{\boldsymbol{\omega}}\right) + \hat{\mathbf{f}}(\mathbf{k},t), \tag{4}$$

where $\bar{\mathbf{u}}$ and $\bar{\boldsymbol{\omega}}$ are the resolved velocity and vorticity in physical space, respectively. Compared with the one for DNS given by Eqs. (3), the governing equation for LES contains an additional eddy-viscosity ν_e, which is calculated using the spectral eddy-viscosity model [52,53] as

$$\nu_e\left(k\,|k_c\right) = \nu_e^+\left(k\,|k_c\right)\sqrt{E\left(k_c\right)/k_c}, \tag{5}$$

$$\nu_e^+\left(k\,|k_c\right) = C_k^{-3/2}\left[0.441 + 15.2\exp\left(-3.03\,k_c/k\right)\right], \tag{6}$$

where k_c denotes the wavenumber corresponding to the filtering scale, $E\left(k_c\right)$ is the energy spectrum value at $k = k_c$, and $C_k = 2.0$ is a constant.

2.3. Filtered direct numerical simulation

To generate low-resolution FDNS data, the DNS velocity field is filtered spatially using the following convolution

$$\tilde{\mathbf{u}}(\mathbf{x}) = \int G(\mathbf{r})\,\mathbf{u}(\mathbf{x}-\mathbf{r})\mathrm{d}\mathbf{r}, \tag{7}$$

where G is the filtering function. In the present study, three types of filter functions listed in Table 2 are used to obtain different groups of FDNS data. In the table, $\tilde{\Delta}$ denotes the filtering scale, with $k_{cf} = \pi/\tilde{\Delta}$ being the corresponding filtering wavenumber. The filtering operation is performed in Fourier space, and the filtered velocity $\tilde{\mathbf{u}}(\mathbf{x},t)$ in physical space is computed as the Fourier transformation of $\hat{\tilde{\mathbf{u}}}(\mathbf{k},t)$. Among the three filter functions listed in Table 2, the sharp spectral filter is sharp in spectral space but nonlocal in physical space; the box filter is local in physical space but not effective at filtering the energy at high wavenumbers; only the Gaussian filter is reasonably compact in both physical and spectral space. The difference of transfer function between

Table 2
Three filter functions used for generated the FDNS data.

Filter function	FDNS velocity $\tilde{\mathbf{u}}(\mathbf{x},t) = \sum_{\lvert\mathbf{k}\rvert=1}^{k_{\max}} \hat{\tilde{\mathbf{u}}}(\mathbf{k},t)\,e^{i\mathbf{k}\cdot\mathbf{x}}$
Sharp spectral filter	$\hat{\tilde{\mathbf{u}}}(\mathbf{k},t) = \hat{\mathbf{u}}(\mathbf{k},t)\cdot H\left(k_{cf} - \lvert\mathbf{k}\rvert\right)$
Gaussian filter	$\hat{\tilde{\mathbf{u}}}(\mathbf{k},t) = \hat{\mathbf{u}}(\mathbf{k},t)\cdot \exp\left(-\frac{\lvert\mathbf{k}\rvert^2\tilde{\Delta}^2}{24}\right)$
Box filter	$\hat{\tilde{\mathbf{u}}}(\mathbf{k},t) = \hat{\mathbf{u}}(\mathbf{k},t)\cdot \frac{\sin\left(\lvert\mathbf{k}\rvert\,\tilde{\Delta}/2\right)}{\lvert\mathbf{k}\rvert\,\tilde{\Delta}/2}$

Table 3
The parameters of DNS at different Re_λ and the generated low-resolution FDNS data with $k_{cf} : k_{\max} = 1 : 4$.

Re_λ	N^3 (DNS)	N^3 (FNDS)	$\tilde{\Delta}/L$
64.43	128^3	32^3	0.2945
128.78	256^3	64^3	0.1473
205.51	512^3	128^3	0.0736
302.04	1024^3	256^3	0.0368

them is compared by the energy spectra of FDNS, as shown in Figure 1 of a previous study [30].

Note that the filtered velocity field obtained from the approach described above has the same resolution as DNS. To extract low-resolution data, we extract velocity at every r grid points in each direction, where $r = k_{\max}/k_{cf}$ denotes the resolution ratio between the FDNS and DNS data. To obtain the flow data with different resolutions, we generate FDNS data with $k_{cf} : k_{\max} = 1 : 4$ for different cases. The parameters of FDNS data are summarized in Table 3.

3. Turbulence volumetric super-resolution model

3.1. Method

To reconstruct 3D high-resolution turbulence data $\mathbf{I}_{HR}$ from low-resolution turbulence data $\mathbf{I}_{LR}$ using a deep learning algorithm, it is essential to establish a mapping function $\mathcal{M}$ between them, viz.

$$\mathbf{I}_{HR} = \mathcal{M}\left(\mathbf{I}_{LR}, \mathbf{w}\right), \tag{8}$$

where $\mathbf{w}$ is the weight. The value of $\mathbf{w}$ is trained using the Adam optimizer [54], an algorithm for gradient-based optimization of stochastic objective functions:

$$\mathcal{L} = \arg\min_{\mathbf{w}}\left(\|\hat{\mathbf{I}}_{HR} - \mathbf{I}_{HR}\|_2^2 + \lambda\|\mathbf{w}\|_2^2\right), \tag{9}$$

where $\hat{\mathbf{I}}_{HR}$ is the high-resolution data reconstructed using the neural network, and the mean square error between $\hat{\mathbf{I}}_{HR}$ and the corresponding DNS data $\mathbf{I}_{HR}$ is used as the loss function. The L2 regularization term $\lambda\|\mathbf{w}\|_2^2$ (λ is an regularization parameter) is included to avoid overfitting. Specifically, the value of $\lambda\|\mathbf{w}\|_2^2$ increases as the neural network becomes more complex. Therefore, this term is added to prevent the neural network from being overly complex, and as such overfitting is avoided.

Fig. 1 shows the schematic diagram of the proposed TVSR network, while the hyperparameters are listed in Table 4. From Fig. 1, it is seen that the proposed TVSR network is mainly composed of residual blocks and upscale modules. The residual network, consisting of multiple residual blocks has exhibited excellent performance in computer vision problems in recently years [55–58]. The number of residual blocks m is an important hyperparameter of the TVSR model. Its effect on the training accuracy is given in Section 3.2. Each residual block consists of two 3D convolutional layers and one rectified linear unit (ReLU) activation layer as shown in Fig. 1(b), and there is a short skip connection between the input and output of the first and second convolutional layers, respectively. Each convolutional layer consists of multiple filter kernels, which is used to extract low-resolution input features. The number of

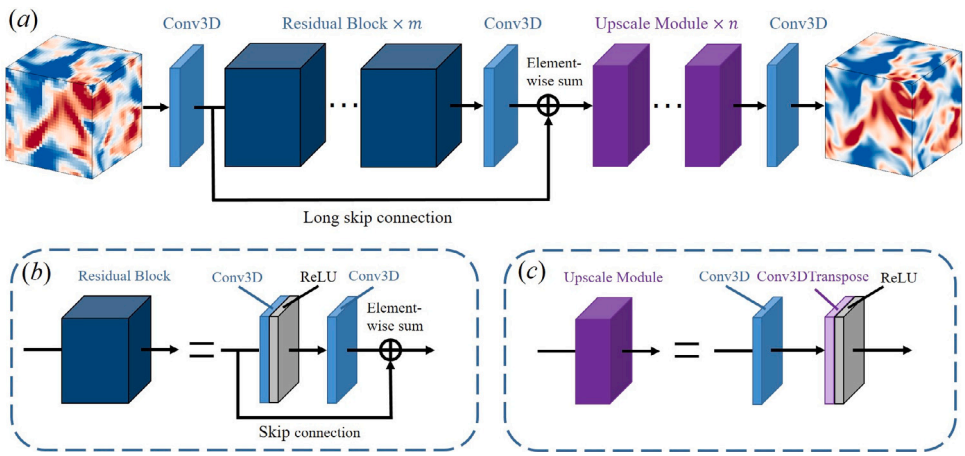

Fig. 1. Schematic diagram of (a) the proposed turbulence volumetric super-resolution network (TVSR) and (b) residual block, (c) upscale module in the TVSR.

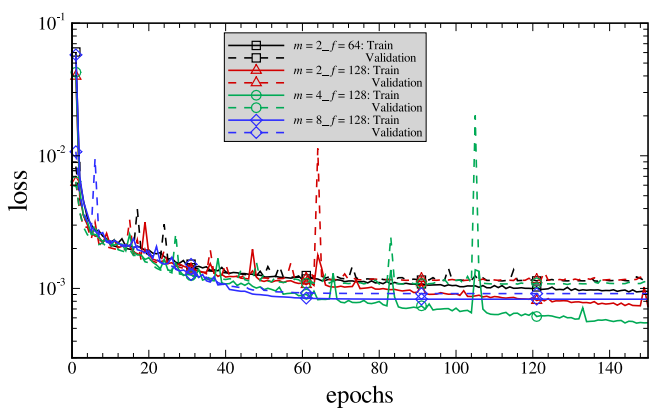

Fig. 2. The variations of the normalized loss function with the training epochs in the training and validation datasets. The loss function is normalized using the mean square of the real output, and the initial error at the zeroth epoch of about 1.0 for both training and validation datasets are not shown in the logarithmic coordinates. The maximum number of training epochs is set to 150.

Table 4
The hyperparameters of the TVSR network.

Number of residual blocks (m)	8
Number of filter kernels (f)	128
Filter kernel size (k)	3
Number of upscaling modules (n)	2
Activation function	ReLU

filter kernels is another adjustable hyperparameter, of which the effect on the training accuracy is also investigated in Section 3.2. In order to avoid the gradient vanishing and exploding problems in the deep network, a long skip connection is applied over the residual blocks, and this treatment is found to be effective at improving the accuracy of the super-resolution model in previous studies using EDSR and RCAN methods [58,59]. The size of filter kernel in each convolutional layer is $k \times k \times k$. Theoretically, increasing the value of k can expand the range of feature extraction from the data, while the computational cost increases correspondingly. The present value $k = 3$ is chosen following the common practice in many previous studies.

The upscale module is composed of one 3D convolutional layer and one 3D transposed convolutional layer [60], followed by a ReLU activation layer as shown in Fig. 1(c). The number of upscale module is determined by the grid ratio r between the low-resolution and high-resolution data. Specifically, each upscale module can upsample the low-resolution input by an upscaling factor $r' = 2$. In the present study, TVSR models for $r = 4$ are trained and tested, and therefore the number of upscale module is $n = 2$. We further note here that although n is a hyperparameter, it is not adjustable but determined by the grid ratio r. In other words, if the grid ratio r changes, a new model needs to be trained. An alternative chain-upscaling strategy can avoid training a new model, that is, to train a model with $r = 2$ using data at different grid resolutions and Reynolds numbers. After training, the model is then applied n times to conduct the super-resolution reconstruction with grid ratio of 2^n. However, the accuracy of the TVSR model based on such a chain-upscaling strategy decreases with the increase of n because of the cumulative error in deep network. In the *a priori* test with $r = 4$, we find that the performance of applying twice the TVSR model with $r = 2$ is much less satisfactory than that of applying the TVSR model with $r = 4$. Therefore, the chain-upscaling strategy is not adopted in the present study, and the corresponding results are not presented in this paper.

3.2. Training of TVSR model

Table 5 gives the parameters of training dataset, including the sizes of input and output data, and the number of training samples. As shown, the flow data at two lower Reynolds numbers, i.e. $Re_\lambda = 64.43$ and 128.78, are used for training the model. To ensure that the model can be generalized to flows at higher Reynolds numbers with different numbers of grid points, we choose a patch-by-patch reconstruction strategy. Such a patchwise reconstruction strategy is also supported by an important property of turbulence, that is, the correlation of velocity at two points diminishes as the separation is sufficiently large. The patch size of $N_p = 16$ is chosen to train the model, and as such the total number of grid points in a 3D patch is 16^3 and 64^3 for FDNS and DNS data, respectively. Here, the "patch size" refers to the number of grid points of each patch. The effects of the patch size on the training accuracy and model performance are investigated in Appendix A. To minimize the discontinuity in the velocity gradient at the edges of two adjacent patches, the patches are chosen with overlapping regions. Specifically, the grid interval between two sampling patches is 4 and 8 for $Re_\lambda = 64.43$ and $Re_\lambda = 128.78$, respectively. Due to the limitation in the computer memory, the data are grouped into batches to train the model. The number of patches in each batch (named as "batch size") is 16, which means that the input of TVSR model for each training step is a matrix with the size of $16 \times 16 \times 16 \times 16$ and the output is a matrix of $64 \times 64 \times 64 \times 16$.

Fig. 2 compares the training loss of TVSR models with different combinations of the numbers of residual block (m) and filter kernel (f). Here, different filter kernel could extract different feature maps, which gives a comprehensive representation of the flow features. The

Table 5
The parameters of our training dataset.

Re_λ	FDNS	DNS	Input patch	Output patch	Grid interval	Number of patches
64.43	32^3	128^3	16^3	64^3	4	2625
128.78	64^3	256^3	16^3	64^3	8	7203

vertical axis represents the loss function normalized by the mean square of the real output $\mathbf{I}_{HR}$, the horizontal axis represents the number of training epochs. In each training epoch, the proportion of training dataset contains 90% of the total training patches, while the remaining 10% patches are used as validation dataset.

As shown in the figure, the initial value of the loss function is relatively large, because the trainable parameters in the network are randomly initialized. The value of the loss function in general decreases as the number of epochs increases. In the few beginning epochs, the weight and bias coefficients are trained to optimize the objective function and the loss decreases rapidly. After that, the values of the normalized loss function for both training and validation datasets approach a steady small value of $O(10^{-3})$ in several training epochs, indicating the success of training. It is also seen from Fig. 2 that when the number of residual blocks is set to $m = 2$ or 4, the validation loss converges to a larger value than the training loss. As m increases to 8, the variation in the loss function between the training and validation datasets decreases significantly. Note that close values of the training loss and validation loss are desired, as this is an indication that the training is not overfitted. Therefore, $m = 8$ is chosen in the present study to train the TVSR model. Furthermore, it is seen that the validation loss converges to similar values when the number of filter kernels is set to $f = 64$ and 128. This indicates that increasing the value of f does not improve the training accuracy, and $f = 128$ is sufficient to train the TVSR model.

Thus far, the effects of the numbers of residual blocks m and filter kernels f are discussed. In Appendix B, we further provide an evaluation on the computational complexity of the TVSR model based on different sets of m and f, together with more details of our training tool. Besides, the requirements for carrying out full DNS at various Reynolds numbers are also given for comparison. It is evident from Appendix B that the computational cost of LES+TVSR model is significantly smaller than DNS.

4. Results and discussion

4.1. Accuracy test

To examine the performance of the TVSR models, the instantaneous and statistical results of the reconstructed fields are compared with the DNS results. In this section, we examine the models in the context of two lower Reynolds numbers $Re_\lambda = 64.43$ and 128.78. The data for these two Reynolds numbers are used to train the model. However, the results presented in this section are obtained from the application of the TVSR models to testing data, which have no overlap with the training data. Such a testing approach is a fundamental requirement of machine learning, that is, after training, the model should be applied to testing data varying from the training data. The test results at higher Reynolds numbers are given in Section 4.2 to examine the generalization capability of the TVSR models.

Fig. 3 displays the contours of three velocity components in a 2D slice at $Re_\lambda = 64.43$ obtained from DNS, FDNS and TVSR models. The Gaussian and sharp spectral filters are denoted as 'GF' and 'SF' in the figure, respectively. As described in Section 2.3, we have also trained a TVSR model corresponding to the box filter. Because both Gaussian filter and box filter are localized operators confined in a finite region in physical space, the two corresponding TVSR models yield similar super-resolution reconstruction in terms of both instantaneous and statistical results. Therefore, we use the Gaussian filter as a representative to present the results of these two models. As shown in Fig. 3, compared to the DNS results in the middle column, the FDNS flow fields shown in the first and last columns are smeared due to the loss of small-scale information in the flow field. The reconstructed flow fields displayed in the second and fourth columns are visually consistent with the DNS results.

To further conduct a quantitative examination of the reconstructed flow field, the energy spectra obtained from DNS, FDNS and TVSR models are compared in Fig. 4. As shown in Fig. 4(a), the energy spectra obtained from the TVSR model corresponding to the Gaussian filter agrees well with the DNS results, indicating that velocity fluctuations at different scales are well reconstructed. From Fig. 4(b), it is seen that although the TVSR model corresponding to the sharp spectral filter underestimates the energy spectra, it mostly recovers the energy at unresolved scales in comparison to the FDNS data. Different from the Gaussian and box filters, the sharp spectral filter is sharp in spectral space, associated with a global dependency in physical space [30,61]. To be specific, the velocity at each single point of the FDNS flow field relies on the velocity of the entire DNS flow field. The proposed TVSR model reconstructs the flow field using a patch-by-patch strategy, while in each patch, the flow data outside the patch remain unknown. Therefore, compared to the FDNS data based on the Gaussian and box filters, it is more difficult for the proposed TVSR model to reconstruct a high-resolution flow field from an FDNS flow field based on a sharp spectral filter.

A straightforward approach for overcoming the aforementioned difficulty in the present patch-by-patch model is to train the model using the entire field data, a strategy that is used in most previous super-resolution models [41,42,47]. However, the global training approach limits the generalization capability of the model to higher Reynolds numbers, because the number of grid points for the low-resolution flow data varies with the Reynolds number. In other words, if a model trained using low-Reynolds number data is used to reconstruct a flow field at a higher Reynolds number, it can be only applied in a patch-by-patch manner. However, different from the velocity fluctuations at inertial subrange, of which the energy spectra follow the $-5/3$ law with respect to the wavenumber, the velocity fluctuations at large scales, or integral scales, do not show similarity as the Reynolds number increases. Therefore, it should not be expected that a model trained using global flow data can be well generalized to a higher Reynolds number. In fact, to the knowledge of the authors, there is no examination of the generalization capability in previous studies of super-resolution models. On the other hand, the generalization capability is a highly desired feature of a super-resolution model, which enables the application of a model trained using flow data at low Reynolds numbers (usually available from DNS) to reconstruct high-resolution data at high Reynolds numbers (infeasible to obtain from DNS due to the limitation in computer power). Based on the above considerations, we decide to choose the patch-by-patch strategy to train the TVSR models, with an acceptable regression in the accuracy.

4.2. Applications to higher Reynolds numbers

In this section, we further investigate the generalization capability of the TVSR models by examining their performances at higher Reynolds numbers ($Re_\lambda = 205.51$ and 302.04), of which the data are not used to train the model. Fig. 5 shows the velocity component u of 3D isotropic turbulence at $Re_\lambda = 302.04$ obtained from FDNS, DNS, KS model and TVSR model corresponding to the Gaussian filter. The results obtained from another super-resolution method, namely the KS model, is also displayed for comparison. In each panel, the upper row shows

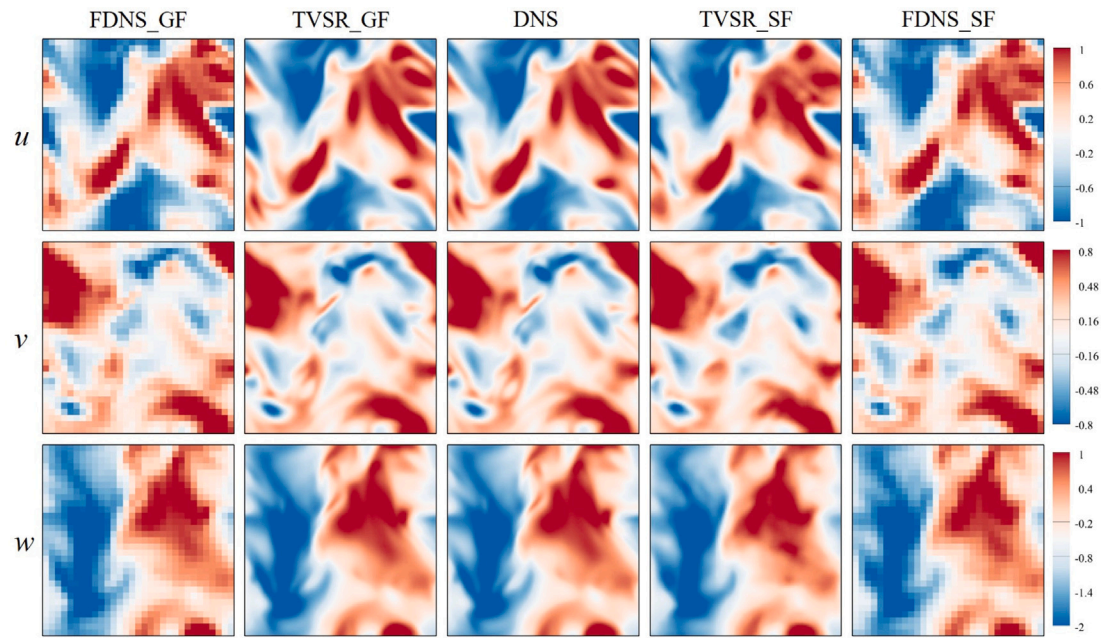

Fig. 3. Contours of the velocity field u, v, w at the $x = 0.0$ slice of 3D isotropic turbulence from DNS, FDNS and TVSR model corresponding to the Gaussian and sharp spectral filters ($Re_\lambda = 64.43$). GF and SF stand for Gaussian filter and sharp spectral filter, respectively.

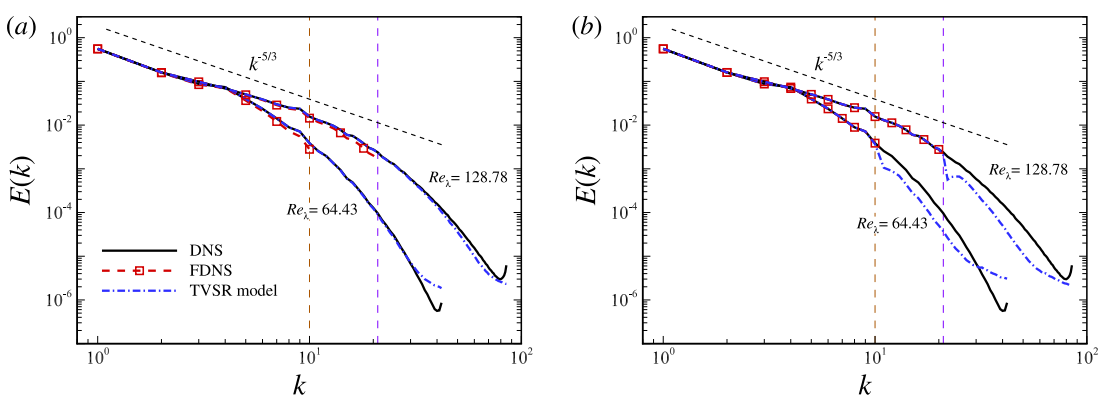

Fig. 4. Energy spectra obtained from DNS, FDNS and TVSR model corresponding to (a) the Gaussian filter and (b) the sharp spectral filter. The results for $Re_\lambda = 64.43$ and 128.78 are shown here.

the entire domain of a 2D slice, while the bottom row displays the left-bottom region, denoted by a dashed box in the upper row. Similar to the results shown in Fig. 3 for the lower Reynolds numbers, the FDNS flow field is smeared, while the reconstructed high-resolution flow field obtained from the TVSR model is visually consistent with the DNS result. In the KS model [23], the subgrid turbulence, generated as a set of random Fourier modes, is superimposed to the low-resolution FDNS flow field. Although the KS model partially recovers the small-scale turbulent structures, the random and frozen subgrid turbulence also makes the plots of the velocity fields noisier than the DNS results. In contrast, the flow field reconstructed using the TVSR model are visually smoother than the results of the KS model.

Fig. 6 compares the energy spectra obtained from DNS, FDNS, TVSR model and KS model. In the KS model, the subgrid energy spectra obtained from DNS are used as input, such that the reconstructed energy spectrum is coincident with the DNS results. In other words, if the DNS result of energy spectra remain unknown, the KS model cannot be used to reconstruct a high-resolution flow field. In contrast, once the training is finished, the TVSR model does not need energy spectra as input to reconstruct the high-resolution flow field due to its generalization capability. As shown in Fig. 6(a), the TVSR model corresponding to the Gaussian filter also gives an accurate prediction of the energy spectra at both $Re_\lambda = 205.51$ and 302.04. The TVSR model corresponding to the sharp spectral filter also makes reasonable prediction on the energy spectra at higher Reynolds numbers in Fig. 6(b), similar to the test results at low Reynolds numbers shown in Fig. 4(b). As noted in Section 4.1, the discrepancy between the results of DNS and TVSR model corresponding to the sharp spectral filter is mainly attributed to the patch-by-patch reconstruction strategy, which however, provides the desirable generalization capability.

Figs. 7 and 8 compare respectively the probability density functions (PDF) of the longitudinal velocity gradient $\partial u/\partial x$ and transverse velocity gradient $\partial u/\partial y$ obtained from DNS, FDNS, TVSR model and KS model. The standard deviations $\sigma_{\partial u/\partial x}$ and $\sigma_{\partial u/\partial y}$ are used to normalize the corresponding velocity gradients. In the FDNS flow field, large-magnitude velocity gradient is filtered out due to the absence of small-scale velocity fluctuations. This is reflected by the relatively small PDF values of large-magnitude velocity gradients in comparison with the DNS results. In the KS model, it is assumed that the small-scale fluctuations follow the Gaussian distribution. As a result, compared to the FDNS results, the PDF values of both longitudinal and transverse velocity gradients obtained from the KS model tend to approach the Gaussian distribution. Figs. 7(a) and 8(a) show that the PDF values of both $\partial u/\partial x$ and $\partial u/\partial y$ obtained from the TVSR model corresponding to

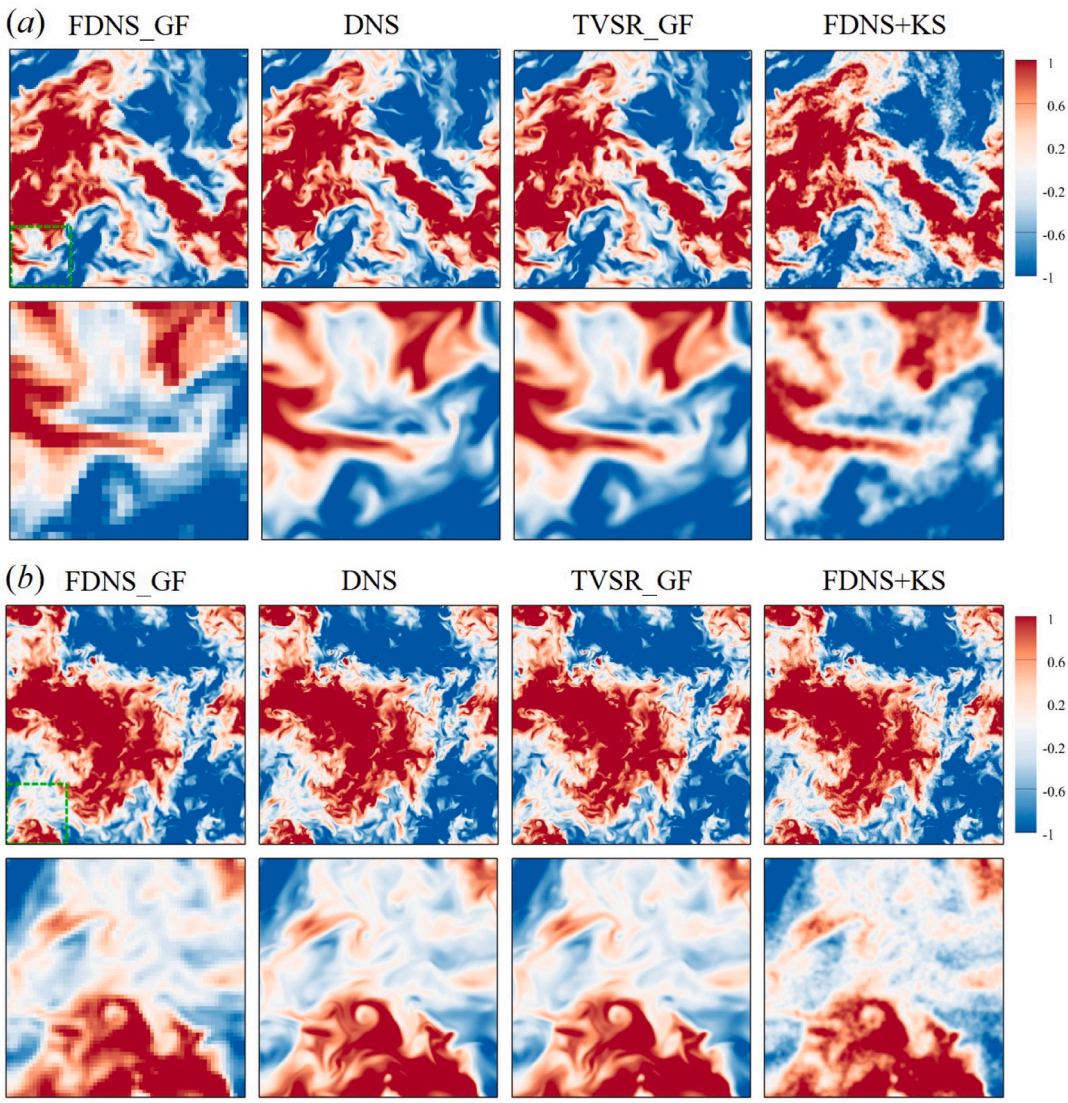

Fig. 5. Contours of the velocity component u obtained from DNS, FDNS, TVSR model and KS model in a 2D slice at (a) $Re_\lambda = 205.51$ and (b) $Re_\lambda = 302.04$. The FDNS data are generated by processing the DNS data using the Gaussian filter. The upper row of each figure shows the entire domain, while the bottom row displays the left-bottom region in a dashed box of the upper row.

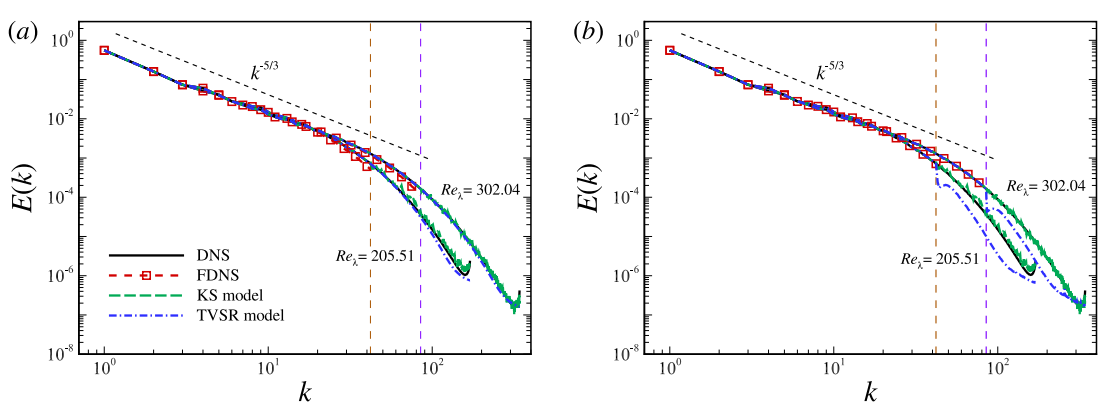

Fig. 6. Energy spectra obtained from the FDNS, DNS, TVSR model and KS model at $Re_\lambda = 205.51$ and 302.04. The FDNS data are generated by processing the DNS data using (a) the Gaussian filter and (b) the sharp spectral filter.

the Gaussian filter agree well with the DNS result. From Figs. 7(b) and 8(b), it is seen that the TVSR model corresponding to the sharp spectral filter only slightly underestimates the PDF values of large-magnitude velocity gradients when compared to the DNS results. As a summary of the investigation of the PDF values of velocity gradients, the TVSR models, corresponding to both Gaussian and sharp spectral filters, make

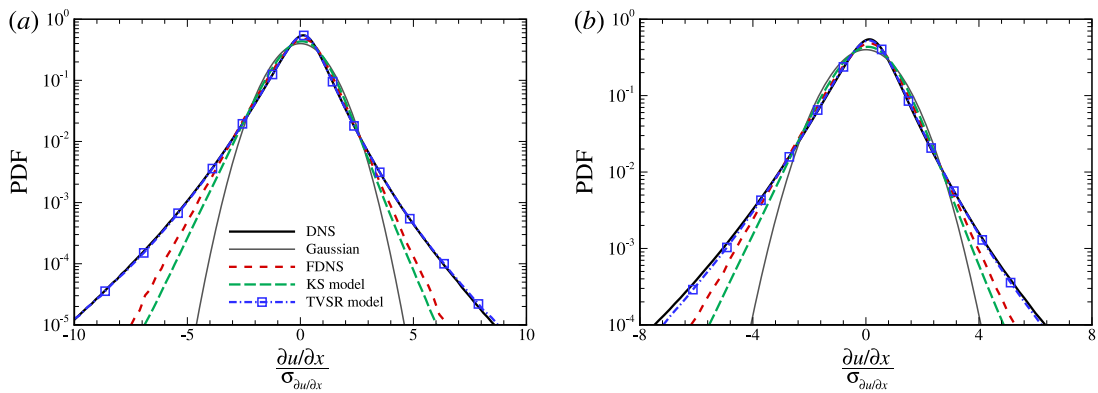

Fig. 7. PDF of the normalized longitudinal velocity gradient obtained from DNS, FDNS, TVSR model and KS model at $Re_\lambda = 302.04$. The FDNS data are generated by processing the DNS data using (a) the Gaussian filter and (b) the sharp spectral filter.

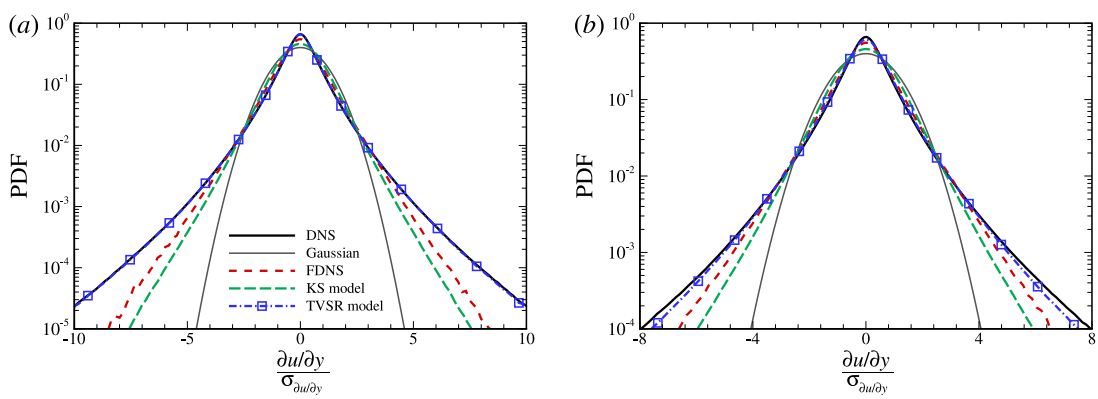

Fig. 8. PDF of the normalized transverse velocity gradient obtained from DNS, FDNS, TVSR model and KS model at $Re_\lambda = 302.04$. The FDNS data are generated by processing the DNS data using (a) the Gaussian filter and (b) the sharp spectral filter. The PDF value is shown in a logarithmic coordinate.

more reasonable reconstruction of small-scale velocity fluctuations than the conventional KS model. Moreover, the incompressibility of the reconstructed velocity fields are preliminarily evaluated based on the data of velocity gradients. For the TVSR model corresponding to Gaussian filter, the continuity equation is satisfied at most computational points.

To evaluate the performance of the TVSR on recovering the vortex structures, we calculate the vortex criterion Q, the second invariant of the velocity gradient tensor. Fig. 9 plots the isosurface of Q obtained from FDNS, DNS, TVSR model and KS model at $Re_\lambda = 205.51$. Compared to the DNS result (Fig. 9a), the small-scale vortex structures are mostly filtered out in the FDNS flow field (Fig. 9b). The KS model (Fig. 9d) reconstructs many small, fragmental vortex structures, which differs significantly from the DNS result. In contrast, the turbulent vortex structures obtained from the TVSR model (Fig. 9c) is visually consistent with the DNS result. To further conduct a quantitative analysis of the performance of the TVSR model on recovering the vortex structures, we calculate the PDF of Q, shown in Fig. 10. It is seen that the PDF of Q obtained from FDNS shows a narrower tail than the DNS result. The KS model partially recovers the PDF of Q. In contrast, the TVSR model corresponding to the Gaussian filter can accurately reconstruct the PDF of Q. The performance of the TVSR model corresponding to the sharp spectral filter is less satisfactory than the TVSR model corresponding to the Gaussian filter, but better than the KS model.

From the results presented in this section, it is seen that the proposed TVSR model shows a generalization capability to higher Reynolds number. This feature is important for a super-resolution model, as it enables the application of the model trained using low-Reynolds-number data (which is relatively easy to obtained due to the smaller computational cost) to reconstruct high-resolution flow fields at higher Reynolds numbers (which are more expensive to be acquired using DNS). In the following content, we further notice the two main reasons that enable the generalization capability of the TVSR model. The first reason is the scale-similarity nature of turbulence motions at small scales. Specifically, the grid size Δ normalized by the Kolmogorov length scale η remain the same in different DNS cases for resolving turbulent motions at all scales (see Section 2.1). Correspondingly, the filtering scale $\widetilde{\Delta}$ of the low-resolution data is also consistent for different Reynolds numbers (see Section 2.3). As such, the velocity around the filtering scale of the low-resolution data is characterized by $\epsilon^{1/3}\eta^{1/3}$, which is used to non-dimensionalize the velocity for model training. In fact, it is well understood in previous studies that if η and $\epsilon^{1/3}\eta^{1/3}$ are used respectively as characteristic length and velocity, the energy spectra for different Reynolds numbers are well scaled at large wavenumbers [2]. The second reason is associated with the patchwise reconstruct strategy. It is understood that the turbulent motions at large scales (or correspondingly, at small wavenumbers) are not scaled by η and $\epsilon^{1/3}\eta^{1/3}$. This means that if more information of large length scale is involved in model training, the effect of Reynolds number becomes stronger. In this sense, the patchwise reconstruction strategy reduces the influence of large-scale turbulent motions to enhance the generalization capability of the TVSR model.

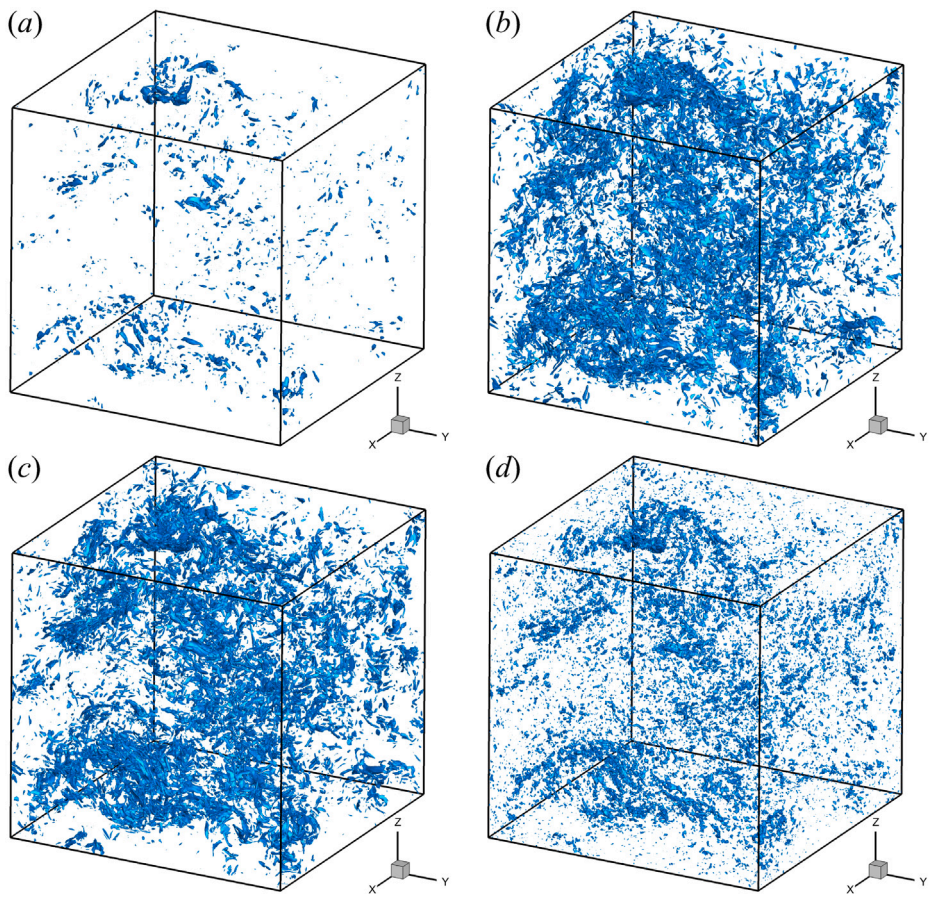

Fig. 9. Isosurface of the Q (= −200) obtained from (a) FDNS, (b) DNS, (c) TVSR model and (d) KS model at $Re_\lambda = 205.51$. The FDNS data are generated by processing the DNS data using the Gaussian filter with $r = 4$.

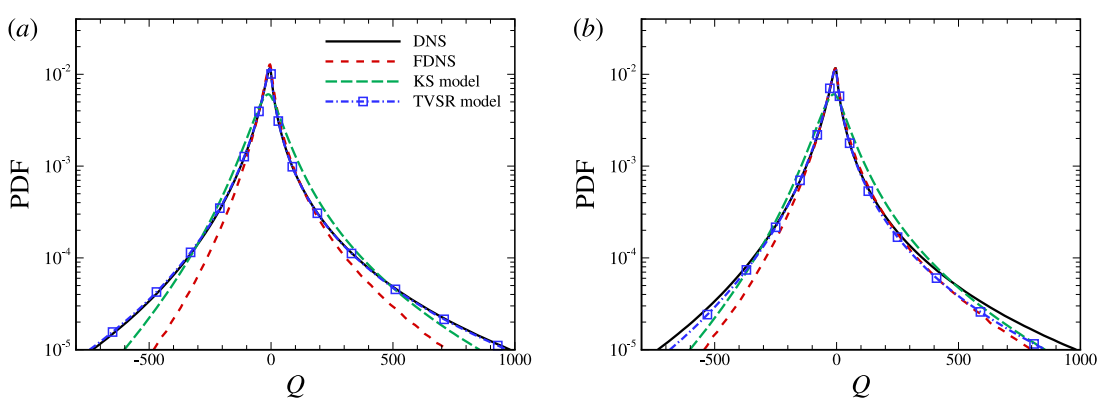

Fig. 10. PDF of the Q obtained from DNS, FDNS, TVSR model and KS model at $Re_\lambda = 302.04$. The FDNS data are generated by processing the DNS data using (a) the Gaussian filter and (b) the sharp spectral filter.

4.3. A posteriori test in LES

Most super-resolution models proposed in previous studies are only examined in the context of processing low-resolution data generated by filtering high-resolution data, as is shown in Section 4.2. However, it is more useful to reconstruct a high-resolution flow field from a low-resolution field obtained directly from an under-resolved simulation, such as LES. The latter task is more challenging in a sense that the DNS data corresponding to the LES data are unavailable. To be more specific, the model can be trained using only FDNS data, of which the corresponding DNS data are known exactly, while after training, it is desired that the model is also robust in reconstructing LES data. In this section, we present the *a posteriori* test results of applying the TVSR model to the reconstruction of high-resolution data obtained from LES data. In the above *a priori* test, we have trained three TVSR models corresponding to different filters. In the *a posteriori* test, the implicit filter of LES is close to a sharp spectral filter in the present simulations based on the pseudo-spectral method. As a result, although the TVSR models corresponding to the Box and Gaussian filters perform well in the *a priori* test, they yield unsatisfactory predictions of the

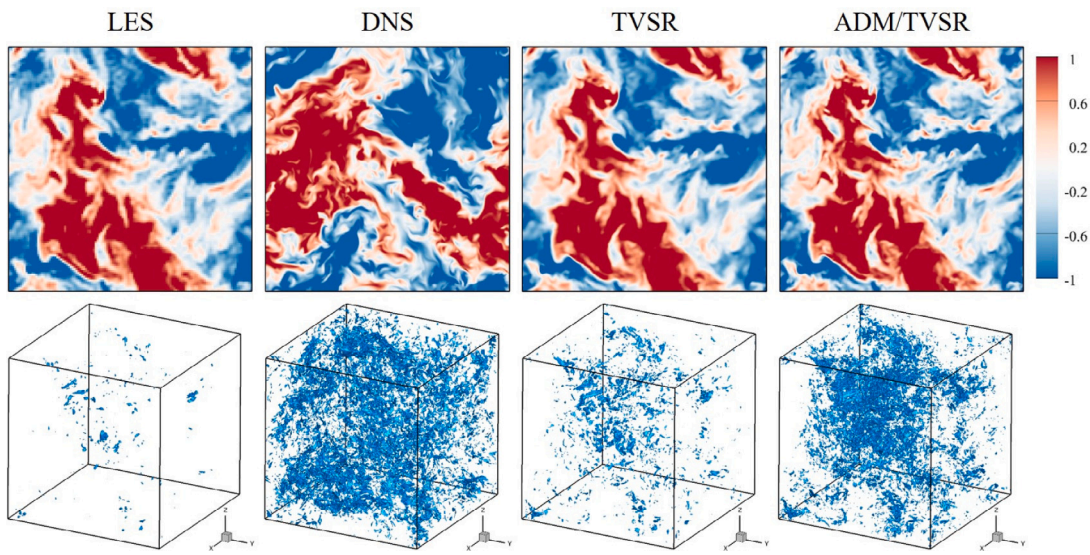

Fig. 11. Contours of the velocity component u in the upper row and isosurfaces of the Q ($= -200$) in the bottom row obtained from LES, DNS, TVSR model corresponding to the sharp spectral filter, and a hybrid ADM/TVSR model at $Re_\lambda = 205.51$.

subgrid energy spectra in the *a posteriori* test. Therefore, the TVSR model corresponding to the sharp spectral filter is chosen to conduct the super-resolution reconstruction of LES flow field.

Fig. 11 plots the contours of velocity component u in the upper row and the isosurfaces of Q in the bottom row obtained from LES, DNS, TVSR model, and a hybrid ADM/TVSR model. Here, the ADM refers to the approximation of the unfiltered solution of LES flow field at resolved scales, which is expressed as

$$\hat{\mathbf{u}}^* = \hat{G}^{-1} \cdot \hat{\bar{\mathbf{u}}}, \tag{10}$$

where $\hat{\bar{\mathbf{u}}}$ is the filtered velocity of LES, $\hat{G}$ is a filter function in spectral space, $\hat{\mathbf{u}}^*$ is an approximation of the unfiltered velocity. The inverse function $\hat{G}^{-1}$ can be approximated using the Taylor series expansion,

$$\begin{aligned}\hat{G}^{-1}(k) &\approx \sum_{n=0}^{N} \left(1 - \hat{G}(k)\right)^n \\ &\approx 6 - 15\hat{G}(k) + 20\hat{G}^2(k) - 15\hat{G}^3(k) + 6\hat{G}^4(k) - \hat{G}^5(k),\end{aligned} \tag{11}$$

where $N = 5$ is sufficient to provide reasonable results for most cases [12]. The filter $\hat{G}(k)$ must satisfy $\left|1 - \hat{G}(k)\right| < 1$ to ensure the existence of its inversion. In this work, we choose the three-dimensional Gaussian transfer function, in which the approximate inverse function increasing monotonously with the wavenumber can strength the motions at smaller scales. Substituting Eq. (11) into Eq. (10), the velocity $\hat{\mathbf{u}}^*$ can be calculated as repeated filtering of $\hat{\bar{\mathbf{u}}}$,

$$\hat{\mathbf{u}}^* = \hat{G}^{-1}\hat{\bar{\mathbf{u}}} \approx 6\hat{\bar{\mathbf{u}}} - 15\hat{\bar{\bar{\mathbf{u}}}} + 20\hat{\bar{\bar{\bar{\mathbf{u}}}}} - 15\hat{\bar{\bar{\bar{\bar{\mathbf{u}}}}}} + 6\hat{\bar{\bar{\bar{\bar{\bar{\mathbf{u}}}}}}} - \hat{\bar{\bar{\bar{\bar{\bar{\bar{\mathbf{u}}}}}}}} \tag{12}$$

According to the previous study [23], the FDNS corresponding to the spherically sharp spectral filter can be regarded as an idealized LES, but the energy near the cut-off scale is significantly underestimated due to the error in the SGS model, while the ADM is able to improve the energy and dissipation spectra near the cut-off scale. In the hybrid ADM/TVSR model, the ADM is utilized first to recover the velocity fluctuations at the resolved scales before the TVSR model is applied. Note that there is no direct correspondence between the DNS and LES instantaneous flow fields. The DNS results are only used as references for a qualitative comparison. It is observed from the first column of Fig. 11 that the velocity field obtained from LES is smeared, while the vortex structures obtained from LES are sparse in space. Shown in the third column, although the TVSR model visually refines the resolution of the LES data, and concentration of small-scale vortex structures (identified by the Q-criterion) obtained from the TVSR model is still lower than the DNS results. As noted above, this is because the energy near the cut-off scale is under estimated in LES. In contrast, the hybrid ADM/TVSR model makes a more reasonable prediction of the high-resolution field, of which the concentration of vortex structures is visually closer to the DNS results.

To further conduct a quantitative examination of the performance of the TVSR model, Fig. 12 shows the energy spectra obtained from DNS, LES, TVSR model, and hybrid ADM/TVSR model. In LES, the energy spectra are absent at high wavenumbers corresponding to subgrid scales and are underestimated near the cut-off wavenumbers. The TVSR model partially recovers the energy spectra at subgrid scales. Compared to the TVSR model, the hybrid ADM/TVSR model makes a better prediction on the energy spectra, because the classical ADM recovers the energy spectrum near the cut-off wavenumbers [12,23]. We note here that it should not be expected that the hybrid ADM/TVSR model can completely recover the energy spectra of DNS for the following reasons. First, the sharp spectral filter is associated with a global correlation in physical space, which to some extent conflicts with the patchwise reconstruction strategy proposed in the present study. However, it is very likely that if the simulation is conducted using other numerical frameworks corresponding to a localized implicit filter, such as finite difference or finite volume methods, this issue can be addressed properly. Second, the error between LES and DNS actually consists of two parts, namely the errors between FDNS and DNS caused by the filter and between LES and FDNS caused by the error associated with the subgrid-scale stress model [62]. Only the error between FDNS and DNS can be treated by the super-resolution model, as it is trained to recover an FDNS field to a DNS field. In the above *a posteriori* test, the ADM partially compensates the deviation between the LES and FDNS fields, but further improvement is still expected. The incorporation of the super-resolution model into the LES solver is a possible approach to give better results [12].

Fig. 13 compares the PDF values of velocity gradients obtained from DNS, LES, ADM, TVSR model, and hybrid ADM/TVSR model. It is seen from Fig. 13 that the ADM imposes little influence on the PDF values of velocity gradients, mainly because it cannot recover any turbulence signals at unresolved scales. The TVSR model makes a good prediction on the PDF of velocity gradients, and the hybrid ADM/TVSR model further improves the prediction.

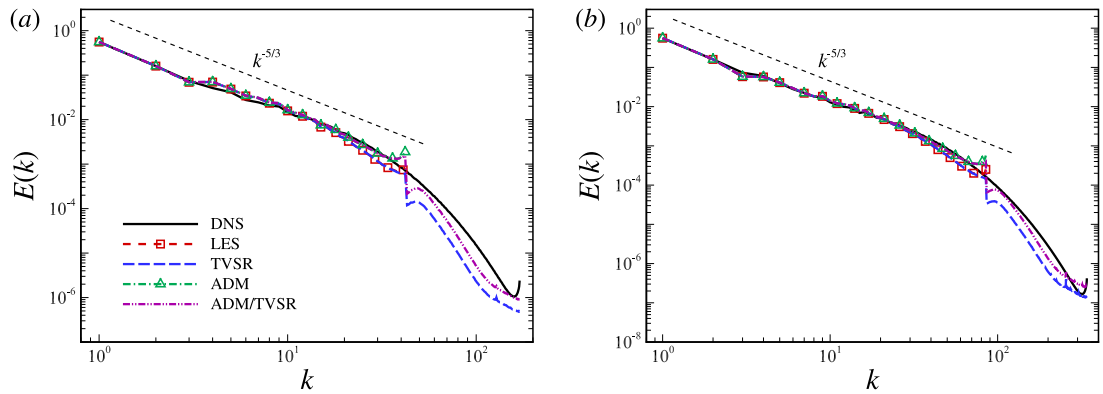

Fig. 12. Energy spectra obtained from DNS, LES, ADM, TVSR model, and hybrid ADM/TVSR model: (a) $Re_\lambda = 205.51$ and (b) $Re_\lambda = 302.04$.

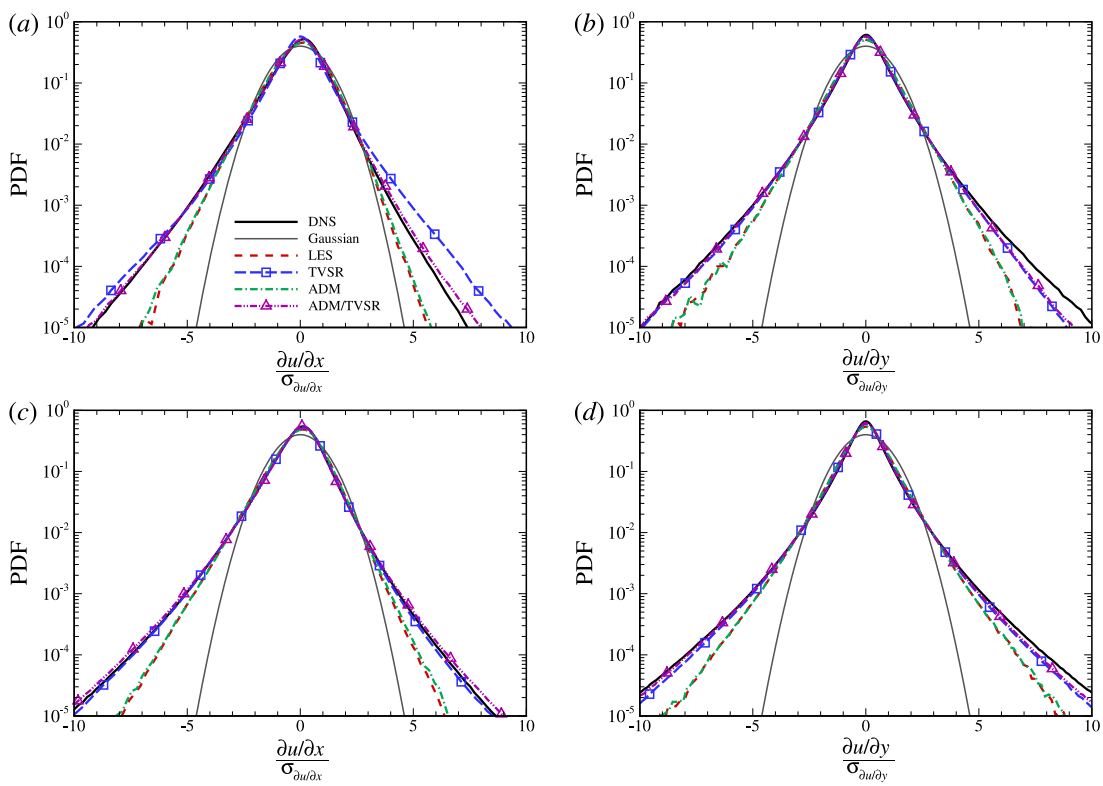

Fig. 13. PDF values of velocity gradients obtained from DNS, LES, ADM, TVSR model, and hybrid ADM/TVSR model for (a, b) $Re_\lambda = 205.51$ and (c, d) $Re_\lambda = 302.04$. Panels (a) and (c) show the PDF values of the longitudinal velocity gradient. Panels (b) and (d) show the PDF values of the transverse velocity gradient.

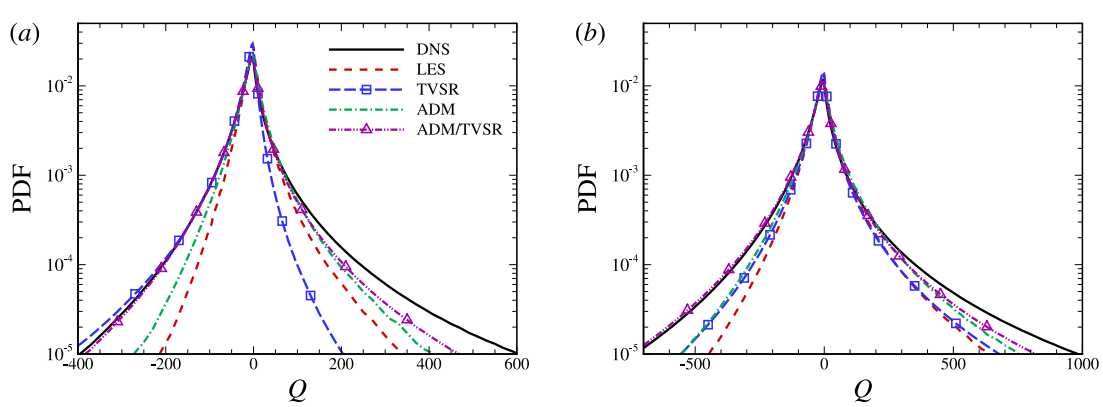

Fig. 14. PDF of the Q obtained from DNS, LES, ADM, TVSR model, and hybrid ADM/TVSR model for (a) $Re_\lambda = 205.51$ and (b) $Re_\lambda = 302.04$.

Fig. 14 compares the PDF of Q obtained from DNS, LES, ADM, TVSR model, and hybrid ADM/TVSR model. As shown, the TVSR model significantly underestimates the PDF value of Q, particularly at large magnitudes, which is even less satisfactory than the ADM. However, the hybrid ADM/TVSR model significantly improves the prediction on the PDF of Q, especially at the negative values. It is understood in previous studies of vortex identifications that negative value of Q is one of the important criteria of vortex structures. In this regard, the hybrid ADM/TVSR model makes a satisfactory reconstruction of the vortex structures, a conclusion that is consistent with the observation from Fig. 11.

5. Conclusion

In this paper, the turbulence volumetric super-resolution (TVSR) model is developed based on convolutional neural network (CNN) to reconstruct the 3D high-resolution turbulent flow data from low-resolution data. To prepare the training samples, three filter functions, namely the box filter, Gaussian filter, and sharp spectral filter, are applied to generate low-resolution FDNS data from high-resolution DNS data. Using the flow data at the two lower Reynolds numbers, the mapping function between low-resolution FDNS data and high-resolution DNS data is established as the TVSR model, which is then applied to reconstruct the high-resolution flow field at various Reynolds numbers.

In the *a priori* test, the TVSR model corresponding to the Gaussian filter (or box filter) makes excellent predictions on both instantaneous flow field and turbulent statistics, including the energy spectra, and PDF values of velocity gradients and invariant Q. The performance of TVSR model corresponding to the sharp spectral filter is not as good as that from the TVSR model corresponding to the Gaussian filter, but it also recovers part of the small-scale turbulence.

In the *a posteriori* test, it is found that when the TVSR model is applied to the super-resolution reconstruction of an LES flow field, it can only recover part of the unresolved energy spectra, and the reconstructed vortex structures are sparser than the DNS result. The hybrid ADM/TVSR model significantly improves the performance over the TVSR model. The concentration of the vortex structures obtained form the hybrid ADM/TVSR model is visually consistent with the DNS result. This observation is quantitatively confirmed through the examination of the PDF of Q. The Hybrid AMD/TVSR model makes a good prediction of the PDF of Q with negative values.

In comparison with existing models based on either conventional method (e.g. the kinematic simulation and approximate deconvolution method) or other machine learning methods (e.g. the fully-connected neural network and hybrid downsampled skip-connection/multi-scale model), the TVSR model shows the following advantages. First, unlike the 2D image super-resolution network that is used in previous models for reconstructing the flow data, the TVSR model directly processes 3D flow field data. As such, the correlations of velocity in all three directions are involved in the TVSR model. Second, the proposed TVSR model shows a good generalization capability in a sense that the model trained using flow data at low Reynolds numbers can be used for super-resolution reconstruction of flow fields at higher Reynolds numbers. The generalization capability is attributed to the patch-by-patch reconstruction strategy. Finally, the TVSR models coupled with the ADM are found robust and accurate in super-resolution reconstruction of flow data obtained from LES.

As a final remark of this paper, we note that the present work is the first step towards the super-resolution reconstruction of LES data of turbulent flows. To further improve the capacity of TVSR model, some key issues need to be addressed. First, the implicit form of filter in LES is unknown and more filter functions, i.e. the general filter, Cauchy filter, Smagorinsky filter and their composite forms, can be implemented into the training [61]. Second, the coupling of data assimilation [63] and machine learning is a promising method to enhance the reconstruction of turbulent flows because the reconstructed results can provide a feedback to the training process. Finally, the present model is limited to the prediction of spatial statistics of turbulence, which can be extended to reconstruct space–time correlations and spectra from incomplete information [64].

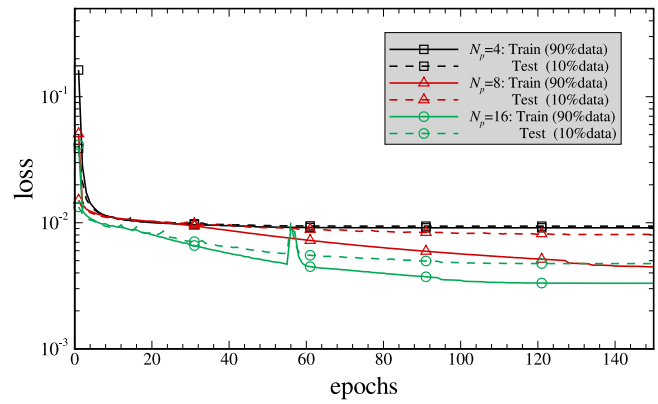

Fig. A.15. The comparison of the normalized loss function for TVSR model with different number of grid points of patch.

CRediT authorship contribution statement

Zhideng Zhou: Conceptualization, Methodology, Investigation, Writing – original draft, Visualization. **Binglin Li:** Software, Validation. **Xiaolei Yang:** Writing – review & editing, Supervision. **Zixuan Yang:** Conceptualization, Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (NSFC) Basic Science Center Program for "Multiscale Problems in Nonlinear Mechanics" (No. 11988102), NSFC project (No. 11972038, 12002345), National Key Project (Grant No. GJXM92579), and the Strategic Priority Research Program (Grant No. XDB22040104). The authors also thank Professor Guowei He for fruitful discussions on simulation methods and modeling strategies.

Appendix A. The effect of patch size on the accuracy of TVSR model

In the proposed patchwise reconstruction strategy, the accuracy of TVSR model is influenced by the patch size. Theoretically, increasing the patch size yields a higher training accuracy of the model by involving more information of flow field in the training. In this appendix, we compare the TVSR models trained with different patch sizes of $N_p = 4$, 8 and 16 to provide a quantitative estimation of the effect of patch size. The flow data at $Re_\lambda = 64.43$ and 128.78 corresponding to sharp spectral filter are used for training, and the parameters of our training datasets with different patch sizes are shown in Table A.6. Other hyperparameters remain the same as those given in Table 4.

Fig. A.15 compares the normalized loss function for TVSR model with different patch sizes. As seen, both the training loss and validation loss decrease as the patch size increases. To further examine the effect of the patch size, the models are applied to reconstruct high-resolution flow data at $Re_\lambda = 205.51$.

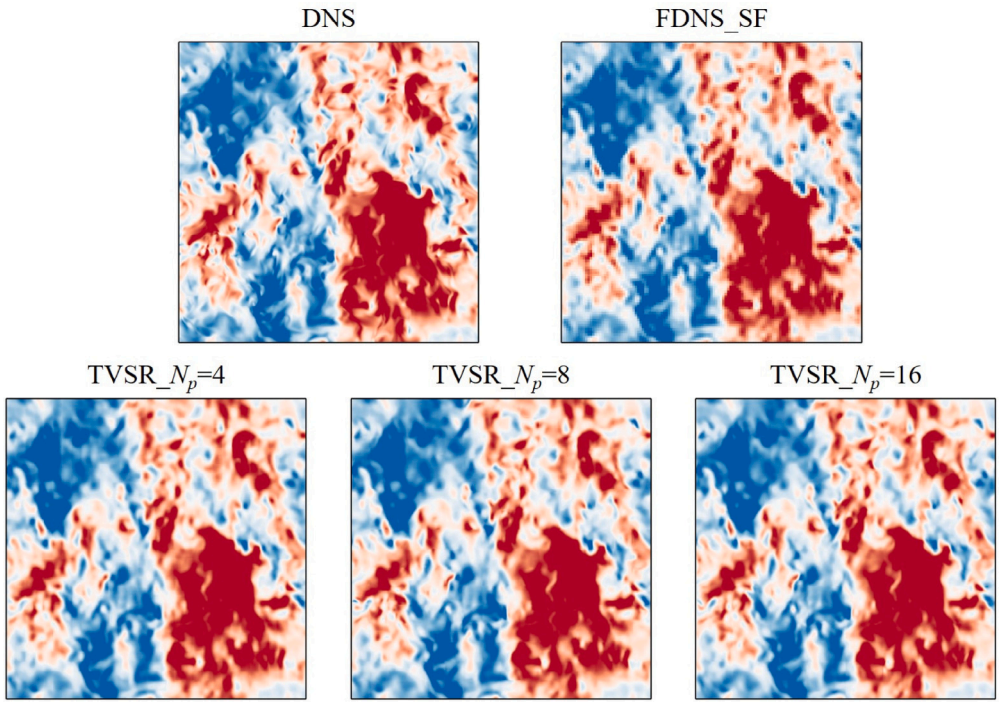

Fig. A.16. Contours of the velocity component u at the $x = 0.0$ slice of 3D isotropic turbulence from DNS, FDNS_SF and TVSR model with different number of grid points of patch ($Re_\lambda = 205.51$).

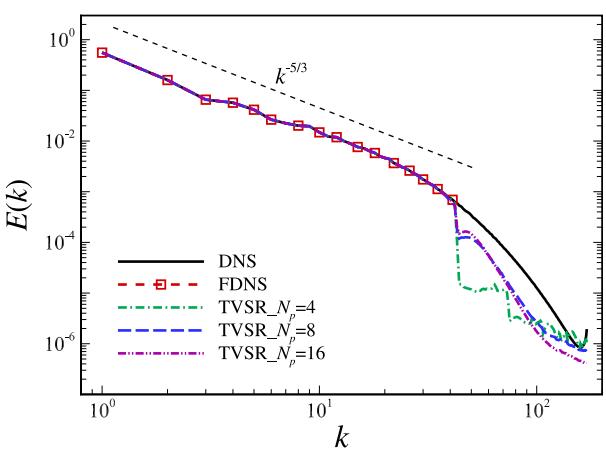

Fig. A.17. The Comparison of energy spectra obtained from DNS, FDNS_SF and TVSR model with different number of grid points of patch ($Re_\lambda = 205.51$).

Table A.6
The parameters of our training datasets with different number of grid points of patch.

Input patch	Output patch	Number of patches	Batch size
4^3	16^3	70875 + 525611	1024
8^3	32^3	7203 + 70875	128
16^3	64^3	2625 + 7203	16

Figs. A.16~A.18 compares the contours of velocity component u, the energy spectra and PDF values of velocity gradients obtained from TVSR models with different patch sizes. The DNS and FDNS results are also shown for validation. In Fig. A.16, the three TVSR models give similar visualization of the velocity contours. However, as shown in Fig. A.17, the energy spectrum of the high-resolution field obtained from the TVSR model with $N_p = 4$ are less satisfactory than the results for $N_p = 8$ and 16. The TVSR model with $N_p = 16$ gives better result of energy spectrum near the cut-off wavenumber than that with $N_p = 8$. Similarly, the PDF values of longitudinal and transverse velocity gradients for $N_p = 4$ is unsatisfactory, while $N_p = 8$ and $N_p = 16$ give reasonable predictions of the PDF of velocity gradients.

Appendix B. Computational costs of TVSR model

In this appendix, we briefly compare the computational costs for implementing the DNS/LES and training the TVSR model. Table B.7 compares the computational costs of different methods. The DNS and LES were performed on the "Tianhe-I" at the National Super Computer Center in Tianjin, China, and the model of the Central Processing Unit (CPU) is Intel Xeon X5670. As shown, the computational cost spend on the development of statistically stationary isotropic turbulence exhibits an exponential growth with the increasing Reynolds number.

The TVSR models were trained on a personal computer, which loads a Graphic Processing Unit (GPU) with the model of NVIDIA RTX2080. The training datasets consisting of 9828 pairs of input (16^3) and output (64^3), are obtained by filtering the DNS flow data at two lower Reynolds numbers, $Re_\lambda = 64.43$ and 128.78. Increasing the complexity of CNN by using a larger number of trainable parameters can improve the accuracy of the model, but meanwhile leads to an increase in the time cost for model training. Once the TVSR model training is accomplished, the turbulent flows at higher Reynolds numbers can be reconstructed based on the low-resolution data.

Taking the case of $Re_\lambda = 302.04$ as an example, the DNS costs 107.2 h and 256 CPUs on developing the statistically stationary turbulent flow field, but only 5.61 h on 64 CPUs are needed to obtain the LES data, and at most 19.1 h on a personal computer are used to train the TVSR model. The super-resolution model is devoted to obtain the high-resolution turbulent data with less time and computational cost.

The code used for training TVSR model is opened in the website "https://github.com/zhouturb/3D-Tubulence-Super-Resolution".

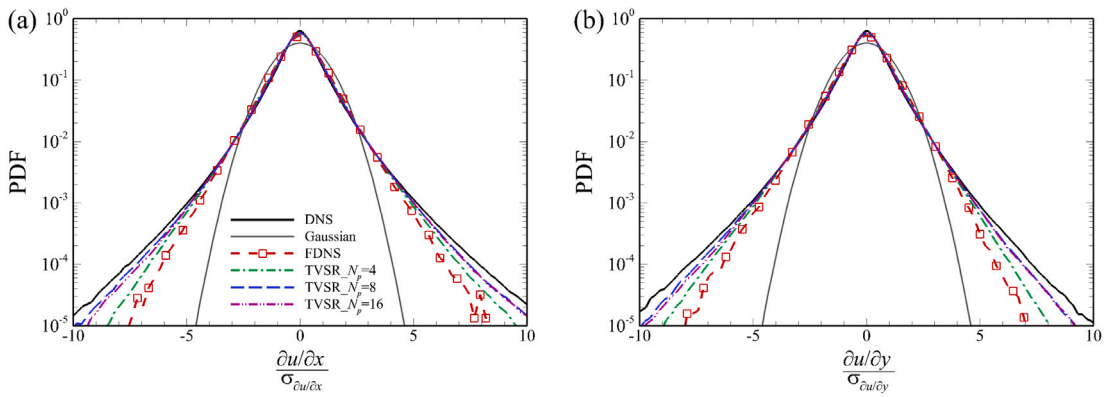

Fig. A.18. PDF values of (a) longitudinal and (b) transverse velocity gradients obtained from DNS, FDNS_SF and TVSR model with different number of grid points of patch ($Re_\lambda = 205.51$).

Table B.7
The computational costs of time for the DNS, LES and the TVSR model.

	N^3	Re_λ	CPU number	Computing time (h)	Cost (CPU · h)
DNS	128^3	64.43	16	0.3	4.8
	256^3	128.78	64	1.9	121.6
	512^3	205.51	128	15.3	1958.4
	1024^3	302.04	256	107.2	27443.2
LES	32^3	64.43	16	0.0025	0.04
	64^3	128.78	16	0.04	0.64
	128^3	205.51	16	1.05	16.8
	256^3	302.04	64	5.61	372.5
TVSR	m	f	Number of variables	Training time/epoch (s)	Total time (h)
	2	32	416497	123	5.1
	2	64	1330193	172	7.2
	2	128	4650577	317	13.2
	4	32	527217	131	5.5
	4	128	6420561	366	15.3
	8	32	748657	158	6.6
	8	128	9960529	458	19.1

References

[1] Moin P, Mahesh K. Direct numerical simulation: A tool in turbulence research. Annu Rev Fluid Mech 1998;30:539–78. http://dx.doi.org/10.1146/annurev.fluid.30.1.539.

[2] Ishihara T, Gotoh T, Kaneda Y. Study of high-Reynolds number isotropic turbulence by direct numerical simulation. Annu Rev Fluid Mech 2009;41:165–80. http://dx.doi.org/10.1146/annurev.fluid.010908.165203.

[3] Choi H, Moin P. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. Phys Fluids 2012;24:011702. http://dx.doi.org/10.1063/1.3676783.

[4] Lesieur M, Métais O. New trends in large-eddy simulations of turbulence. Annu Rev Fluid Mech 1996;28:45–82. http://dx.doi.org/10.1146/annurev.fl.28.010196.000401.

[5] Meneveau C, Katz J. Scale-invariance and turbulence models for large-eddy simulation. Annu Rev Fluid Mech 2000;32:1–32. http://dx.doi.org/10.1146/annurev.fluid.32.1.1.

[6] Armenio V, Piomelli U, Fiorotto V. Effect of the subgrid scales on particle motion. Phys Fluids 1999;11:3030. http://dx.doi.org/10.1063/1.870162.

[7] He G-W, Rubinstein R, Wang L-P. Effects of subgrid-scale modeling on time correlations in large eddy simulation. Phys Fluids 2002;14:2186. http://dx.doi.org/10.1063/1.1483877.

[8] Yang Y, He G-W, Wang L-P. Effects of subgrid-scale modeling on Lagrangian statistics in large eddy simulation. J Turbul 2008;9:N8. http://dx.doi.org/10.1080/14685240801905360.

[9] Jin GD, He G-W, Wang L-P. Large-eddy simulation of turbulent collision of heavy particles in isotropic turbulence. Phys Fluids 2010;22:055106. http://dx.doi.org/10.1063/1.3425627.

[10] He GW, Jin GD, Yang Y. Space-time correlations and dynamic coupling in turbulent flows. Annu Rev Fluid Mech 2017;49:51–70. http://dx.doi.org/10.1146/annurev-fluid-010816-060309.

[11] Stolz S, Adams NA. An approximate deconvolution procedure for large-eddy simulation. Phys Fluids 1999;11:1699. http://dx.doi.org/10.1063/1.869867.

[12] Stolz S, Adams NA, Kleiser L. An approximate deconvolution model for large-eddy simulation with application to incompressible wall-bounded flows. Phys Fluids 2001;13:997. http://dx.doi.org/10.1063/1.1350896.

[13] Cernick MJ, Tullis SW, Lightstone MF. Particle subgrid scale modelling in large-eddy simulations of particle-laden turbulence. J Turbul 2014;16(2):101–35. http://dx.doi.org/10.1080/14685248.2014.969888.

[14] Das SK, Durbin PA. A Lagrangian stochastic model for dispersion in stratified turbulence. Phys Fluids 2005;17:025109. http://dx.doi.org/10.1063/1.1849184.

[15] Mazzitelli IM, Toschi F, Lanotte AS. An accurate and efficient Lagrangian sub-grid model. Phys Fluids 2014;26:095101. http://dx.doi.org/10.1063/1.4894149.

[16] Heppe BMO. Generalized Langevin equation for relative turbulent dispersion. J Fluid Mech 1998;357:167–98. http://dx.doi.org/10.1017/S0022112097008069.

[17] Barge A, Gorokhovski MA. Acceleration of small heavy particles in homogeneous shear flow: Direct numerical simulation and stochastic modelling of under-resolved intermittent turbulence. J Fluid Mech 2020;892:A28. http://dx.doi.org/10.1017/jfm.2020.162.

[18] Kraichnan RH. Diffusion by a random velocity field. Phys Fluids 1970;13:22. http://dx.doi.org/10.1063/1.1692799.

[19] Fung JCH, Hunt JCR, Malik NA, Perkins RJ. Kinematic simulation of homogeneous turbulence by unsteady random Fourier modes. J Fluid Mech 1992;236:281–318. http://dx.doi.org/10.1017/S0022112092001423.

[20] Fung JCH, Vassilicos JC. Two-particle dispersion in turbulentlike flows. Phys Rev E 1998;57:1677–90. http://dx.doi.org/10.1103/PhysRevE.57.1677.

[21] Flohr P, Vassilicos JC. A scalar subgrid model with flow structure for large-eddy simulations of scalar variances. J Fluid Mech 2000;407:315–49. http://dx.doi.org/10.1017/S0022112099007533.

[22] Yao H-D, He G-W. A kinematic subgrid scale model for large-eddy simulation of turbulence-generated sound. J Turbul 2009;10:1–14. http://dx.doi.org/10.1080/14685240903032725.

[23] Zhou ZD, Wang SZ, Jin GD. A structural subgrid-scale model for relative dispersion in large-eddy simulation of isotropic turbulent flows by coupling kinematic simulation with approximate deconvolution method. Phys Fluids 2018;30:105110. http://dx.doi.org/10.1063/1.5049731.

[24] Zhou ZD, Wang SZ, Yang XL, Jin GD. A structural subgrid-scale model for the collision-related statistics of inertial particles in large-eddy simulations of isotropic turbulent flows. Phys Fluids 2020;32:095103. http://dx.doi.org/10.1063/5.0018756.

[25] Ghate AS, Lele SK. Subfilter-scale enrichment of planetary boundary layer large eddy simulation using discrete Fourier-Gabor modes. J Fluid Mech 2017;819:494–539. http://dx.doi.org/10.1017/jfm.2017.187.
[26] Ghate AS, Lele SK. Gabor mode enrichment in large eddy simulations of turbulent flow. J Fluid Mech 2020;903:A13. http://dx.doi.org/10.1017/jfm.2020.622.
[27] Duraisamy K, Iaccarino G, Xiao H. Turbulence modeling in the age of data. Annu Rev Fluid Mech 2019;51:357–77. http://dx.doi.org/10.1146/annurev-fluid-010518-040547.
[28] Brunton SL, Noack BR, Koumoutsakos P. Machine learning for fluid mechanics. Annu Rev Fluid Mech 2020;52:477–508. http://dx.doi.org/10.1146/annurev-fluid-010719-060214.
[29] Ling J, Kurzawski A, Templeton J. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. J Fluid Mech 2016;807:155–66. http://dx.doi.org/10.1017/jfm.2016.615.
[30] Zhou ZD, He GW, Wang SZ, Jin GD. Subgrid-scale model for large-eddy simulation of isotropic turbulent flows using an artificial neural network. Comput Fluids 2019;195:104319. http://dx.doi.org/10.1016/j.compfluid.2019.104319.
[31] Zhou ZD, He GW, Yang XL. Wall model based on neural networks for LES of turbulent flows over periodic hills. Phys Rev Fluids 2021;6:054610. http://dx.doi.org/10.1103/PhysRevFluids.6.054610.
[32] King R, Hennigh O, Mohan A, Chertkov M. From deep to physics-informed learning of turbulence: Diagnostics. 2018, arXiv:181007785v2.
[33] Lee S, You D. Data-driven prediction of unsteady flow over a circular cylinder using deep learning. J Fluid Mech 2019;879:217–54. http://dx.doi.org/10.1017/jfm.2019.700.
[34] Wu Z, Lee J, Meneveau C, Zaki T. Application of a self-organizing map to identify the turbulent-boundary-layer interface in a transitional flow. Phys Rev Fluids 2019;4:023902. http://dx.doi.org/10.1103/PhysRevFluids.4.023902.
[35] Li BL, Yang ZX, Zhang X, He GW, Deng B-Q, Shen L. Using machine learning to detect the turbulent region in flow past a circular cylinder. J Fluid Mech 2020;905:A10. http://dx.doi.org/10.1017/jfm.2020.725.
[36] Zhou Y, Fan DW, Zhang BF, Li RY, Noack BR. Artificial intelligence control of a turbulent jet. J Fluid Mech 2020;897:A27. http://dx.doi.org/10.1017/jfm.2020.392.
[37] Park J, Choi H. Machine-learning-based feedback control for drag reduction in a turbulent channel flow. J Fluid Mech 2020;904:A24. http://dx.doi.org/10.1017/jfm.2020.690.
[38] Maulik R, San O. A neural network approach for the blind deconvolution of turbulent flows. J Fluid Mech 2017;831:151–81. http://dx.doi.org/10.1017/jfm.2017.637.
[39] Maulik R, San O, Rasheed A, Vedula P. Data-driven deconvolution for large eddy simulations of Kraichnan turbulence. Phys Fluids 2018;30:125109. http://dx.doi.org/10.1063/1.5079582.
[40] Beck A, Flad D, Munz C-D. Deep neural networks for data-driven LES closure models. J Comput Phys 2019;398:108910. http://dx.doi.org/10.1016/j.jcp.2019.108910.
[41] Fukami K, Fukagata K, Taira K. Super-resolution reconstruction of turbulent flows with machine learning. J Fluid Mech 2019;870:106–20. http://dx.doi.org/10.1017/jfm.2019.238.
[42] Deng ZW, He CX, Liu YZ, Kim KC. Super-resolution reconstruction of turbulent velocity fields using a generative adversarial network-based artificial intelligence framework. Phys Fluids 2019;31:125111. http://dx.doi.org/10.1063/1.5127031.
[43] Wang XT, Yu K, Wu SX, Gu JJ, Liu YH, Dong C, et al. ESRGAN: Enhanced super-resolution generative adversarial networks. 2018, arXiv:180900219v2.
[44] Bode M, Gauding M, Kleinheinz K, Pitsch H. Deep learning at scale for subgrid modeling in turbulent flows: Regression and reconstruction. 2019, arXiv:191000928v1.
[45] Bode M, Gauding M, Lian ZY, Denker D, Davidovic M, Kleinheinz K, et al. Using physics-informed super-resolution generative adversarial networks for subgrid modeling in turbulent reactive flows. 2019, arXiv:191111380v1.
[46] Fukami K, Nabae Y, Kawai K, Fukagata K. Synthetic turbulent inflow generator using machine learning. Phys Rev Fluids 2019;4:064603. http://dx.doi.org/10.1103/PhysRevFluids.4.064603.
[47] Liu B, Tang JP, Huang HB, Lu XY. Deep learning methods for super-resolution reconstruction of turbulent flows. Phys Fluids 2020;32:025105. http://dx.doi.org/10.1063/1.5140772.
[48] Xie CY, Wang JC, Li H, Wan MP, Chen SY. Spatially multi-scale artificial neural network model for large eddy simulation of compressible isotropic turbulence. AIP Adv 2020;10:015044. http://dx.doi.org/10.1063/1.5138681.
[49] Kim H, Kim J, Won S, Lee C. Unsupervised deep learning for super-resolution reconstruction of turbulence. J Fluid Mech 2021;910:A29. http://dx.doi.org/10.1017/jfm.2020.1028.
[50] Chen JC, Jin GD, Zhang J. Lagrangian statistics in isotropic turbulent flows with deterministic and stochastic forcing schemes. Acta Mech Sin 2015;31:25–31. http://dx.doi.org/10.1007/s10409-015-0012-x.
[51] Canuto C, Hussaini MY, Quarteroni A, Jr. TAZ. Spectral methods in fluid dynamics. Berlin, Heidelberg: Springer-Verlag; 1988, http://dx.doi.org/10.1007/978-3-642-84108-8.
[52] Chollet JP, Lesieur M. Parameterization of small scales of three-dimensional isotropic turbulence utilizing spectral closure. J Atmos Sci 1981;38:2747–57. http://dx.doi.org/10.1175/1520-0469(1981)038<2747:POSSOT>2.0.CO;2.
[53] Chollet JP. Two-point closure used for a sub-grid scale model in large eddy simulations. In: Bradbury LJS, Durst F, Launder BE, Schmidt FW, Whitelaw JH, editors. Turbulent shear flows 4. Berlin, Heidelberg: Springer-Verlag; 1985, http://dx.doi.org/10.1007/978-3-642-69996-2_5.
[54] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv:14126980v9.
[55] Dong C, Deng YB, Loy CC, Tang XO. Compression artifacts reduction by a deep convolutional network. In: 2015 IEEE international conference on computer vision. Santiago, Chile; 2015, p. 576–84. http://dx.doi.org/10.1109/ICCV.2015.73.
[56] He K, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition. Las Vegas, NV; 2016, p. 770–8. http://dx.doi.org/10.1109/CVPR.2016.90.
[57] Kim J, Lee JK, Lee KM. Accurate image super-resolution using very deep convolutional networks. In: 2016 IEEE conference on computer vision and pattern recognition. Las Vegas, NV; 2016, p. 1646–54. http://dx.doi.org/10.1109/CVPR.2016.182.
[58] Zhang Y, Li K, Li K, Wang L, B. Z, Fu Y. Image super-resolution using very deep residual channel attention networks. 2018, arXiv:180702758v2.
[59] Lim B, Son S, Kim H, Nah S, Lee KM. Enhanced deep residual networks for single image super-resolution. 2017, arXiv:170702921v1.
[60] Dong C, Loy CC, He KM, Tang XO. Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 2016;38:295–307. http://dx.doi.org/10.1109/TPAMI.2015.2439281.
[61] Pope SB. Turbulent flows. Cambridge: Cambridge University Press; 2000, http://dx.doi.org/10.1017/CBO9780511840531.
[62] Li R, Yang B, Yang Z, Wang S, He G. Error of large-eddy simulation in the wall pressure fluctuation of a turbulent channel flow. Theor Appl Mech Lett 2021;11:100248. http://dx.doi.org/10.1016/j.taml.2021.100248.
[63] Zhang X-L, Xiao H, He G-W, Wang S-Z. Assimilation of disparate data for enhanced reconstruction of turbulent mean flows. Comput Fluids 2021;224:104962. http://dx.doi.org/10.1016/j.compfluid.2021.104962.
[64] Wu T, He GW. Local modulated wave model for the reconstruction of space-time energy spectra in turbulent flows. J Fluid Mech 2020;886:A11. http://dx.doi.org/10.1017/jfm.2019.1044.