





RESEARCH ARTICLE | OCTOBER 17 2024

Application of convolutional neural network for efficient turbulence modeling in urban wind field simulation

Rui Zhao (赵睿) ; Siyang Zhong (钟思阳) ; Ruoyu You (尤若于)  



Physics of Fluids 36, 105169 (2024)

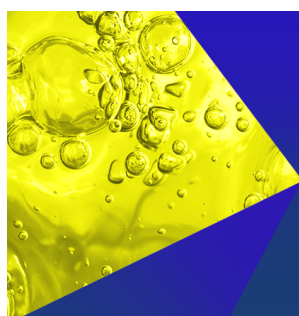
<https://doi.org/10.1063/5.0233053>



Articles You May Be Interested In

Energy consumption, international trade, and real income in the USA: An empirical investigation using conditional error correction models

J. Renewable Sustainable Energy (December 2014)



Physics of Fluids
Special Topics
Open for Submissions

[Learn More](#)

Application of convolutional neural network for efficient turbulence modeling in urban wind field simulation

Cite as: Phys. Fluids **36**, 105169 (2024); doi: 10.1063/5.0233053

Submitted: 12 August 2024 · Accepted: 2 October 2024 ·

Published Online: 17 October 2024



View Online



Export Citation



CrossMark

Rui Zhao (赵睿),¹ Siyang Zhong (钟思阳),² and Ruoyu You (尤若于)^{1,a)}

AFFILIATIONS

¹Department of Building Environment and Energy Engineering, The Hong Kong Polytechnic University, Kowloon 999077, Hong Kong SAR, China

²Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Kowloon 999077, Hong Kong SAR, China

^{a)} Author to whom correspondence should be addressed: ruoyu.you@polyu.edu.hk

ABSTRACT

Accurate flow field estimation is crucial for the improvement of outdoor environmental quality, but computational fluid dynamics (CFD) based on the widely used Reynolds-averaged Navier–Stokes method has limitations in this regard. This study developed a turbulence modeling framework based on a convolutional neural network (CNN) to model turbulence in urban wind fields. The CNN model was trained by learning the Reynolds stress patterns and spatial correlations with the use of high-fidelity datasets. Next, the model was integrated into the CFD solver to generate accurate and continuous flow fields. The generalization capability of the proposed framework was initially demonstrated on the simplified benchmark configurations. The validated framework was then applied to case studies of urban wind environments to further assess its performance, and it was shown to be capable of delivering accurate predictions of the velocity field around an isolated building. For more complex geometries, the proposed framework performed well in regions where the flow properties were covered by the training dataset. Moreover, the present framework provided a continuous and smooth velocity field distribution in highly complicated applications, underscoring the robustness of the proposed turbulence modeling framework.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0233053>

I. INTRODUCTION

Modern lifestyles have led individuals to spend the majority of their time indoors.¹ However, outdoor activities are still important as they are associated with a reduction in the morbidity of diabetes,² Parkinsons' disease,³ and depression.⁴ Therefore, outdoor environmental quality is essential for supporting healthier and more active lifestyles.⁵ A critical factor in outdoor environmental quality is wind distribution,⁶ which influences thermal comfort by affecting heat dissipation and air circulation around people.⁷ Furthermore, wind distribution plays an important role in ventilation within urban areas and influences the dispersal and dilution of airborne pollutants.⁸ Therefore, it is important to predict outdoor wind distribution accurately and effectively.

With the increase in numerical advances and computational powers, computational fluid dynamics (CFD) based on either Reynolds-averaged Navier–Stokes (RANS) solutions or large eddy simulations (LES) has become popular for studying outdoor wind distribution.⁹ Ideally, LES provides more accurate results and small-scale

flow structures than RANS, but the computational cost is much higher.¹⁰ Moreover, in practice, the LES method typically entails stringent requirements for boundary conditions and numerical schemes, which hinder its practical applicability.^{11–13} For example, Vita *et al.*¹¹ reported that LES performed worse than RANS in the absence of appropriate inlet turbulence, the accurate modeling of which is a long-standing challenge in CFD.^{14,15} Consequently, the LES method is still not suitable for complex urban environments where precise boundary conditions and detailed inflow turbulence data are difficult to acquire. In contrast, the RANS method is computationally efficient and requires only a simple setup, making it preferable for many industrial applications, including urban wind simulations. However, the method is still lacking in the ability to predict the outdoor wind velocity accurately, especially for complex urban geometries.^{16–18} Shirzadi and Tominaga¹⁶ conducted both RANS and LES simulations to study the wind distribution around a high-rise building surrounded by street canyons. They found that the normalized root mean square error for the wind velocity calculated by RANS was 63% higher than the error

for the LES-calculated velocity in the upwind cavity. Furthermore, Liu *et al.*¹⁸ studied the wind distribution around scaled building complexes and reported that the velocity calculated by the RANS method exhibited an average difference of 40% compared to the wind tunnel data. Also, most RANS models rely on the Boussinesq hypothesis,¹⁹ which may lead to inaccurate predictions of complex outdoor flow patterns, including separated flows and anisotropy flows.²⁰ For example, RANS models were unable to correctly reproduce a reattachment and recirculation zone on a roof surface and behind a building.^{21,22} For inaccurate RANS predictions, uncertainty quantification has been used to estimate the potential error range of the RANS results.^{23,24} Although uncertainty quantification provides a more reliable interpretation of the RANS predictions, it does not inherently improve the accuracy of the model. Inherent limitations of RANS models, such as model simplification and bias, still lead to incorrect predictions of key flow characteristics. Therefore, it is necessary to develop methods that improve the ability of RANS to accurately predict outdoor wind distribution.

Many attempts have been made to improve the RANS method. One commonly adopted measure is the development of targeted semi-empirical corrections in the turbulent transport equations to account for the major properties of the specific flows under investigation.^{25–29} For example, Engblom *et al.*²⁶ introduced a dynamic correction factor to increase the diffusion of turbulent kinetic energy (TKE) k and turbulent dissipation rate ε , thus providing a significant improvement in the RANS prediction of jet flows. Smirnov and Menter²⁷ applied a rotation factor to the shear stress transport model of RANS to account for system rotation and streamline curvature. Kumar and Dewan²⁹ developed a buoyancy-modified version of the k - ε model by incorporating a corrected production term into the turbulent transport equations, and the modified model exhibited good agreement with experimental measurements for turbulent thermal plumes. Although these corrections have improved RANS prediction accuracy under specific circumstances, they cannot be immediately applied to complex scenarios due to the differences among specific flow types. In recent years, some researchers have explored the use of data-driven methods, such as machine learning, to modify turbulence models.^{30–33} The advantage of this approach is that the correction can be developed on the basis of a large amount of data, allowing it to be applied to a wide range of flows. For example, Yan *et al.*³⁰ employed field inversion and machine learning to correct the Spalart–Allmaras model³⁴ and found that the enhanced model could be generalized to flow conditions similar to those of the training dataset. Therefore, the data-driven method is promising for improving the accuracy of RANS predictions.

Data-driven methods have proven to be effective in improving the accuracy of RANS predictions.^{35–41} Note that data-driven methods have also been applied to develop surrogate models to establish end-to-end mappings from input parameters.⁴² However, the primary purpose of these models is to reduce computational resource consumption rather than enhance accuracy. Therefore, the following review focuses on the enhancement of RANS accuracy through data-driven methods. The most common method for this purpose is to predict the Reynolds stress, which is further incorporated into the CFD solver to compute the flow variables. Ling *et al.*³⁵ pioneered a tensor-basis neural network (TBNN) to develop new Reynolds stress closures based on the tensor representation theory derived by Pope,⁴³ enabling the prediction of anisotropic flows. The results showed that in comparison to the traditional RANS model, a TBNN based on a fully connected neural

network reproduced the anisotropic flows in the cases of duct flow case and a wavy wall and provided more accurate velocity predictions with improvements of 44% and 56%, respectively. Kaandorp *et al.*³⁹ developed a similar framework based on the random forest (RF) method and introduced a Gaussian filter to improve the smoothness of the predictions. The method provided good predictions of the reattachment locations of the backward-facing step case, and the resulting velocity profiles agreed well with the accurate but expensive direct numerical simulation (DNS) results. Panda *et al.*⁴⁰ employed gradient boosted decision trees (GBDT) to construct the Reynolds stress anisotropy tensor, which was reported to have better performance than that of the fully connected neural networks and RFs, and the CFD solver was more stable because fewer weights were involved. Although the machine learning models developed in the above research have improved the performance of RANS simulations, these models were trained to predict Reynolds stress at each grid cell independently, without considering spatial correlations or interactions with surrounding grid cells, which can result in discontinuities or a serrated distribution in both the predicted Reynolds stress and final flow distribution.^{39,44} More importantly, this issue may cause convergence problems for the CFD solver. Therefore, it is essential to incorporate the spatial characteristics of the flow field into the model to produce a continuous and accurate flow distribution.

The present study aimed to incorporate the spatial characteristics of the flow field into the machine learning model in order to provide a continuous and accurate outdoor wind distribution. A CNN model was constructed to incorporate the spatial characteristics of the flow field and establish the mapping between the RANS flow field and the high-fidelity Reynolds stress. The CNN model was then integrated with a CFD solver to generate the final flow field. The accuracy and generalization capability of the proposed framework were evaluated by training and testing the model on simplified benchmark datasets. Finally, the validated framework was applied to three case studies of outdoor wind environments to further demonstrate its practicality.

II. METHODOLOGY

A. Theoretical foundation: Algebraic Reynolds stress model

The continuity equation and momentum equation for incompressible flows are as follows:⁴⁵

$$\nabla \cdot \mathbf{u} = 0, \quad (1)$$

$$\mathbf{u} \cdot \nabla \mathbf{u} - \nu \nabla^2 \mathbf{u} + \nabla p - \nabla \cdot \boldsymbol{\tau} = 0, \quad (2)$$

where \mathbf{u} is the velocity, ν is the molecular viscosity, p is the mean pressure, and $\boldsymbol{\tau}$ is the so-called Reynolds stress tensor, which must be modeled in RANS.

The commonly used turbulence modeling approaches, such as the k - ε model, rely on the Boussinesq hypothesis, and only provide a low-order approximation of the Reynolds stress. In contrast, the algebraic Reynolds stress model (ARSM) offers a more accurate and higher order representation of the Reynolds stress.⁴⁶ Pope⁴³ proposed the decomposition form of Reynolds stress in the 1970s. For three-dimensional flows, he constructed ten tensor bases using the normalized strain rate tensor \mathbf{s} and rotation rate tensor \mathbf{w} as the tensor parameters, as follows:

$$\begin{cases} T_1 = s, \\ T_2 = \mathbf{sw} - \mathbf{ws}, \\ T_3 = s^2 - \frac{1}{3}I \cdot \text{tr}(s^2), \\ T_4 = \mathbf{w}^2 - \frac{1}{3}I \cdot \text{tr}(\mathbf{w}^2), \\ T_5 = \mathbf{ws}^2 - s^2\mathbf{w}, \end{cases} \begin{cases} T_6 = \mathbf{w}^2\mathbf{s} + \mathbf{sw}^2 - \frac{2}{3}I \cdot \text{tr}(s\mathbf{w}^2), \\ T_7 = \mathbf{ws}\mathbf{w}^2 - \mathbf{w}^2\mathbf{sw}, \\ T_8 = \mathbf{s}\mathbf{w}\mathbf{s}^2 - s^2\mathbf{ws}, \\ T_9 = \mathbf{w}^2\mathbf{s}^2 + \mathbf{s}^2\mathbf{w}^2 - \frac{2}{3}I \cdot \text{tr}(s^2\mathbf{w}^2), \\ T_{10} = \mathbf{ws}^2\mathbf{w}^2 - \mathbf{w}^2s^2\mathbf{w}, \end{cases} \quad (3)$$

where $\mathbf{w} = \frac{1}{2} \frac{k}{\varepsilon} (\nabla \mathbf{u} - \nabla^T \mathbf{u})$, $\mathbf{s} = \frac{1}{2} \frac{k}{\varepsilon} (\nabla \mathbf{u} + \nabla^T \mathbf{u})$, k is the turbulent kinetic energy (TKE), ε is the turbulent dissipation rate, $\text{tr}(\cdot)$ denotes the trace of a tensor, and I is the Kronecker delta appropriate to three dimensions.

Based on Pope's decomposition, the Reynolds stress anisotropy tensor \mathbf{b} can be expressed as a linear combination of these ten tensors,

$$\mathbf{b} = \frac{\tau}{k} - \frac{2}{3}I = \sum_{i=1}^{10} G_i T_i, \quad (4)$$

where G_i are the coefficients of tensor bases. For the three-dimensional flow, these coefficients can be expressed as a function of five invariant features q_1, q_2, \dots, q_5 , which are listed below

$$\begin{aligned} q_1 &= \text{tr}(s^2), & q_2 &= \text{tr}(\mathbf{w}^2), & q_3 &= \text{tr}(\mathbf{w}^3), \\ q_4 &= \text{tr}(\mathbf{w}^2\mathbf{s}), & q_5 &= \text{tr}(\mathbf{w}^2\mathbf{s}^2). \end{aligned} \quad (5)$$

One of the advantages of this decomposition is that it maintains Galilean invariance, which implies that the Reynolds stress calculated

using Eq. (4) remains unchanged when the velocity vector \mathbf{u} undergoes a Galilean transformation. However, it was pointed out that the employment of the five invariants, i.e., q_1, q_2, \dots, q_5 , as independent variables in Pope's decomposition to represent \mathbf{b} may lead to multi-value issues.^{37,47} Therefore, it is necessary to expand the set of independent variables to address this problem. In addition, Eq. (4) is not compact, and any five tensor bases in Eq. (3) are sufficient to represent the Reynolds stress tensor.⁴⁸

B. Data-driven turbulence modeling framework

The general concept of the framework is to employ a machine learning model to predict the Reynolds stress, and the predicted Reynolds stress expressed in the ARSM form can be incorporated to replace the original turbulence model of the RANS simulation. Figure 1 illustrates the framework of machine learning-based turbulence modeling employed in the present study, which is an adaptation of the framework by Ling *et al.*³⁵ The present framework can be summarized as follows: In the model training phase, a machine learning model is constructed using inputs from the RANS-predicted flow field, including features q_i and tensor bases T_i , with $i = 1, \dots, 5$. Moreover, q_i includes the five invariants listed in Eq. (5) and seven additional variables to avoid the multi-value issue (see the discussion later). The outputs include the Reynolds stress anisotropy tensor \mathbf{b} from high-fidelity data, and the TKE discrepancy $\Delta \ln k$, which is the logarithm of the ratio between the high-fidelity TKE k^{HFi} and RANS TKE k^{RANS} , defined as

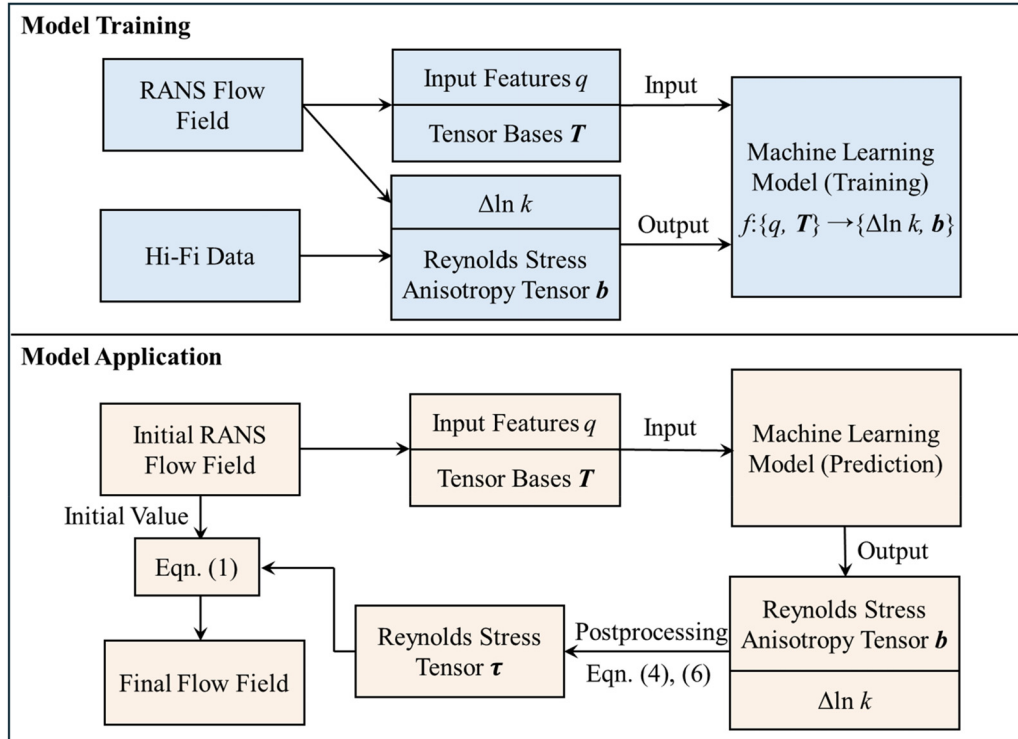


FIG. 1. The machine learning-based turbulence modeling framework developed in the present study.

TABLE I. Input features for the machine learning model employed in the present study.

Feature	Description	Feature	Description
$q_1 = \text{tr}(\mathbf{s}^2)$	Invariant	$q_7 = \frac{2k}{(2k + u_i u_i)}$	Ratio of turbulence/mean kinetic energy
$q_2 = \text{tr}(\mathbf{w}^2)$	Invariant	$q_8 = \frac{\nu_t}{(\nu + \nu_t)}$	Ratio of turbulence/mean viscosity
$q_3 = \text{tr}(\mathbf{s}^3)$	Invariant	$q_9 = \frac{k}{\nu \omega}$	Normalized factor
$q_4 = \text{tr}(\mathbf{w}^2 \mathbf{s})$	Invariant	$q_{10} = \mathbf{w}/(\mathbf{w} + \omega)$	Marker of shear layer and swirler flow
$q_5 = \text{tr}(\mathbf{w}^2 \mathbf{s}^2)$	Invariant	$q_{11} = \min\left(\frac{\sqrt{k}d}{50\nu}, 2\right)$	Wall-distance based Reynolds number
$q_6 = \frac{k}{\varepsilon \ \mathbf{s}\ }$, where $\ \cdot\ $ is the norm of the tensor	Ratio of turbulent timescale to mean strain timescale	$q_{12} = \frac{\hat{\mathbf{u}} \cdot \nabla \hat{p}}{1 + \hat{\mathbf{u}} \cdot \nabla \hat{p} }$, where $\hat{\mathbf{u}} = \frac{\mathbf{u}}{\sqrt{\mathbf{u} \cdot \mathbf{u}}}$, $\nabla \hat{p} = \frac{\nabla p}{\sqrt{(\nabla p) \cdot (\nabla p)}}$	Marker of adverse pressure gradient flow

$$\Delta \ln k = \ln(k^{\text{HiFi}}/k^{\text{RANS}}). \quad (6)$$

In the application phase, the initial flow field is provided by RANS simulation. The features q_j and tensor bases T_i are extracted from this initial flow field and subsequently input to the machine learning model. The model outputs, \mathbf{b} and $\Delta \ln k$, are then post-processed to calculate the Reynolds stress tensor $\boldsymbol{\tau}$ with the use of Eqs. (4) and (6). The predicted Reynolds stress $\boldsymbol{\tau}$ is then incorporated into the RANS flow field to replace that modeled by the original turbulence model. Note that to obtain the final flow field, the turbulence transport equations are disabled, and only Eqs. (1) and (2) are solved. Another point to note is that the current framework is designed for steady RANS, with future developments anticipated to expand its applicability to unsteady RANS.

The input features employed in the present study are listed in Table I. As discussed in Sec. II A, the use of only five invariant features may lead to a multivalued problem. Therefore, more features are needed, and previous studies have recommended many other features that can identify key regions or characteristics of the flow field.^{39,49–51} To ensure that the machine learning model generates a continuous Reynolds stress distribution, we select an additional seven features that have been validated for ideal continuity and smoothness. In addition, the first five tensor bases in Eq. (3) are employed to construct the Reynolds stress tensor, avoiding numerical disturbances arising from the incorporation of higher order tensor bases.

The outputs of the model are post-processed to ensure the realizability of the turbulence. For the Reynolds anisotropy tensor \mathbf{b} , constraints are enforced on the tensor itself and its eigenvalues ξ according to the inequality below.⁵² For components that do not satisfy the inequality, their values are adjusted to the specified bounds. For example, if the predicted $b_{11} > 2/3$, then the upper bound of the inequality can be chosen to update b_{11} so that $b_{11} = 2/3$. Similarly, if the $\xi_1 > 1/3 - \xi_2$, then the ξ_1 is set to $1/3 - \xi_2$, and the tensor \mathbf{b} is then reconstructed,

$$\begin{cases} -\frac{1}{3} \leq b_{ij} \leq \frac{2}{3} & \text{for } j = i, \\ -\frac{1}{2} \leq b_{ij} \leq \frac{1}{2} & \text{for } j \neq i, \end{cases} \quad \begin{cases} \xi_1 \geq (3|\xi_2| - \xi_2)/2, \\ \xi_1 \leq \frac{1}{3} - \xi_2. \end{cases} \quad (7)$$

Moreover, the predicted TKE discrepancy $\Delta \ln k$ is constrained based on the upper and lower bound values observed in the training dataset. It should be noted that this constraint is intended solely to prevent the machine learning model from making unreasonable extrapolated predictions and lacks physical significance.

The calculated Reynolds stress tensor needs to be integrated into the CFD solver so that the eventual flow field can be obtained. However, previous research indicated that explicitly incorporating the Reynolds stress tensor into the CFD solver may not yield the desired flow field and may even cause numerical instability.⁵³ To solve this problem, the Reynolds stress decomposition method proposed by Wu *et al.*⁵⁴ has been adopted in this work. The Reynolds stress is decomposed into linear and nonlinear parts, where the linear part is iterated with the solver implicitly, and the nonlinear part is explicitly added to Eq. (2). It can be expressed as follows:

$$\boldsymbol{\tau} = 2\nu_t^m \mathbf{S} + \boldsymbol{\tau}_\perp, \quad (8)$$

where the term $2\nu_t^m \mathbf{S}$ represents the linear part, $\boldsymbol{\tau}_\perp$ is the nonlinear part, \mathbf{S} is the strain rate tensor, and ν_t^m is the optimal eddy viscosity, for which there are multiple options. In this study, the chosen optimal eddy viscosity is the turbulent viscosity ν_t^{RANS} calculated by RANS, as suggested by Guo *et al.*⁵⁵ With this method, the detailed calculation process is summarized in the algorithm below.

Algorithm 1. Implicit treatment of the Reynolds stress.

- Run RANS simulation to obtain viscosity ν_t^{RANS} and strain rate $\mathbf{S}^{(0)}$
- Run the machine learning model to obtain the corrected Reynolds stress $\boldsymbol{\tau}$
- For** each iteration step of the simulation $i = 1, 2, \dots, n$ do the following:
 - Perform Reynolds stress calculation: $\boldsymbol{\tau} = 2\nu_t^{\text{RANS}} \mathbf{S}^{(i)} + \boldsymbol{\tau}_\perp$
 - Solve $\mathbf{u} \cdot \nabla \mathbf{u} - (\nu + \nu_t^{\text{RANS}}) \nabla^2 \mathbf{u} + \nabla p - \nabla \cdot \boldsymbol{\tau}_\perp = 0$ to obtain updated \mathbf{u}
- End**

This algorithm is an adaptation of Algorithm 2 in Ref. 54. The difference lies in the utilization of RANS viscosity ν_t^{RANS} instead of the calculated viscosity ν_t^m to ensure numerical stability. The proposed algorithm was implemented by means of the simpleFoam solver using the SIMPLE algorithm⁴⁵ in OpenFOAM v2206.⁵⁶

C. Construction of the machine learning model

The turbulence patterns of the flow field exhibit significant spatial interdependencies. However, most current studies have employed machine learning models to predict the Reynolds stress for each grid point independently, without considering the spatial structure and correlation of turbulence. This limitation can lead to inaccurate or physically unrealistic predictions. The CNN model offers a viable solution to identify and learn spatial features through the convolutional layer. Given this, we developed a CNN-based Reynolds stress model for RANS, referred to as CNN-RSM RANS, following the procedure outlined in Sec. II B.

In the present study, the three-dimensional U-Net model,⁵⁷ a special CNN architecture, was employed for turbulence modeling. The network architecture was designed to adopt the concept of TBNN³⁵ to embed Galilean invariance based on Eq. (4). More specifically, the U-Net model processes the input features q_j to predict the coefficients G_i and $\Delta \ln k$. The predicted coefficients are then fed into a merge layer. An additional input layer receives tensor bases T_i and channels them into the same merge layer, which utilizes T_i and G_i to calculate the Reynolds anisotropy tensor b according to Eq. (4).

When the three-dimensional CNN model is used for turbulence modeling, the input data from the flow field needs to be pre-processed. First, the input data are resampled onto uniform grids, as convolutional operation requires structured and uniform grid data as input and output. Also, a set of annotated notations labeled with boundary types is batch inputted into the model to identify the boundary conditions and set the padding methods accordingly. Specifically, circular padding is applied for periodic boundaries, zero padding for wall boundaries, and duplicate padding for zero-gradient conditions.

In the training process, the loss functions are defined as the mean square error (MSE) between outputs and label values,

$$\begin{cases} L_b = \frac{1}{6n} \left(\sum G_i T_i - b_i^{\text{true}} \right)^2 = \frac{1}{6n} \left(b_i^{\text{pred}} - b_i^{\text{true}} \right)^2, \\ L_k = \frac{1}{n} \left(\Delta \ln k^{\text{pred}} - \Delta \ln k^{\text{true}} \right)^2, \end{cases} \quad (9)$$

where the subscript i represents the i th component of the tensor, the superscripts *true* and *pred* represent the true and predicted values, respectively, and n is the total number of data points. In the training phase, two models were independently trained to predict b and $\Delta \ln k$, and the AdamW optimiser⁵⁸ with a learning rate of 0.001 was employed to minimize the MSE losses defined in Eq. (9) to adjust the model parameters. The model performance was evaluated on a validation set that was independent of the training set and test set. To prevent overfitting and ensure the generalization capabilities of the model, an early stopping strategy was employed. Specifically, the training was halted when the MSE on the validation set no longer showed a reduction over 50 consecutive training epochs, which was chosen to prevent overfitting while ensuring sufficient time for the model to learn. Then the model parameters that correspond to the lowest observed MSE on

the validation set were selected as the configuration for the final model.

The model training was conducted on an NVIDIA GeForce RTX 4080 GPU with CUDA version 12.1. The total training duration for a set of models, one predicting b and the other $\Delta \ln k$, was 2.4 h, which included phases of data preprocessing, model initialization, and training. Note that once the model was trained, it could be directly applied across a wide range of environments with significantly reduced computational requirements, as the model only required inference computations at the application phase.

III. VALIDATIONS USING SIMPLIFIED BENCHMARK CONFIGURATIONS

The aim of this study was to develop a turbulence modeling framework for practical outdoor wind distribution prediction. Given the limited availability of high-fidelity training data, it was crucial to ensure the generalization capability of the CNN model. To this end, we initially prepared a dataset containing high-fidelity Reynolds stress data and RANS flow field data. Next, two increasingly challenging tests were designed and executed to assess the performance of the proposed framework prior to its application in real-world scenarios.

A. Dataset

As illustrated in Fig. 2, the complete dataset employed in the present study comprises six distinct flow configurations for which high-fidelity data are available. The dataset includes five periodic hill cases characterized by the steepness ratio α ($Re_H = 5600$),⁵⁹ a curved backward-facing step ($Re_H = 13700$),⁶⁰ five parametric bump cases ($Re_H = 13250$ – 27850),⁶¹ a backward-facing step flow ($Re_H = 5100$),⁶² two converging-diverging channel cases ($Re_H = 12600$ and 20580),⁶³ and 16 square duct cases ($Re_H = 1100$ – 3500).⁶⁴ Note that the dataset also includes the RANS data, which serve as the training input, as detailed in Sec. II B. The RANS data were prepared with the standard k - ε model using the OpenFOAM v2206⁵⁶ as the standard k - ε model has been proven to be robust for a wide range of applications. The detailed settings of the RANS simulations can be found in Refs. 65 and 66.

Two tests were designed to evaluate the performance of the CNN-RSM RANS by employing different training and test datasets, as shown in Table II. A case from the training dataset was chosen as the validation set to identify the optimal moment at which to stop training and avoid overfitting. The rest of the cases were employed as the training set to fit the CNN model by adjusting its parameters.

B. Cross-dataset predictive performance

Figure 3 shows the Reynolds shear stress and velocity field calculated by the baseline RANS using the k - ε model, CNN-RSM RANS, and DNS. Regarding the Reynolds shear stress, as shown in Figs. 3(a)–3(c), the CNN model successfully predicted the general profile and sign of the Reynolds stress with satisfactory smoothness. Although the CNN model was trained on different flow configurations, its predicted Reynolds stress distribution pattern was closer to the DNS results than that predicted by RANS. To illustrate, the CNN model reproduced the low shear stress area behind the left hill, as in the DNS result, whereas the baseline RANS model predicted that the area was primarily located near the right hill. However, the CNN model slightly underpredicted the size of the low shear stress area and overestimated the shear stress magnitude near the inlet and outlet, which probably occurred because

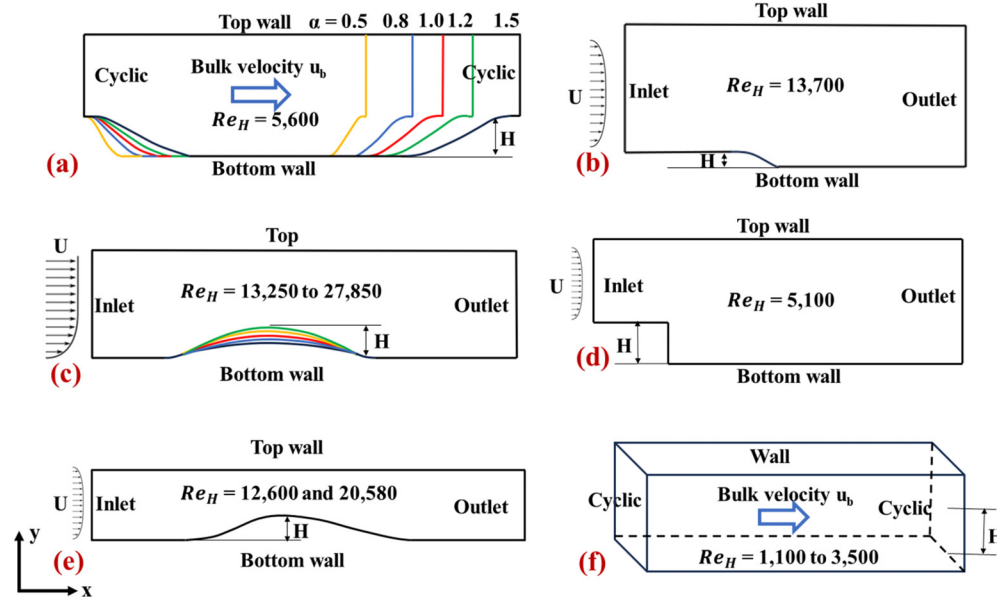


FIG. 2. Flow configurations of the complete high-fidelity dataset for training and testing the CNN model: (a) five periodic hill cases, (b) a curved backward-facing step case, (c) five parametric bump cases, (d) a backward-facing step case, (e) two converging-diverging channel cases, and (f) 16 square duct cases.

the Reynolds number of the training set ($Re_H = 13\,250\text{--}25\,200$) was higher than that of the test case ($Re_H = 5600$). Nevertheless, in light of the final velocity profiles, which are of greater interest in practical applications, the proposed CNN-RSM RANS can provide comparable predictions to those obtained by DNS, as shown in Figs. 3(d)–3(f). The normalized mean absolute error (NMAE) of the velocity was introduced to quantify the error between calculated and DNS velocities, defined as

$$NMAE = \frac{\sum_i |U_{cal}^{(i)} - U_{DNS}^{(i)}| V^{(i)}}{\sum_i V^{(i)}} \bigg/ \frac{\sum_i |U_{DNS}^{(i)}| V^{(i)}}{\sum_i V^{(i)}}, \quad (10)$$

where $V^{(i)}$ is the volume of the i th grid, $U_{cal}^{(i)}$ is the velocity magnitude calculated by baseline RANS or CNN-RSM RANS, and $U_{DNS}^{(i)}$ is the DNS velocity. The CNN-RSM RANS exhibited a substantially lower NMAE, achieving a reduction of 37.93% compared to the baseline RANS. Moreover, the reattachment length provided by CNN-RSM RANS was closely aligned with DNS data. The above

results demonstrated the strong generalization capability of the proposed framework across datasets.

C. Cross-dimensional predictive performance

Figure 4 compares vector plot and velocity components by the baseline RANS using the $k\text{--}\epsilon$ model, CNN-RSM RANS, and DNS. According to the results, the baseline RANS model failed to simulate the secondary flow, as evidenced by the blank vector plot and zero v and w velocity components. In contrast, the secondary flow was observed in the flow field generated by CNN-RSM RANS, although the intensity was lower than the DNS result. This can be interpreted as successful predictions of all Reynolds stress components by the CNN (see the Appendix). Notably, in the prediction of τ_{yz} , the baseline RANS failed, whereas the CNN predicted the correct sign and similar τ_{yz} magnitude to that of the DNS results, despite overpredicting the size of the low shear stress region. In addition, the distribution of the mainstream velocity component u predicted by CNN-RSM RANS was found to be similar to that of the baseline RANS. However, the u velocity calculated by CNN-RSM RANS exhibited a smaller NMAE,

TABLE II. Test configurations for evaluation of CNN-RSM RANS performance.

Test	Training dataset		Test dataset	Description
	Training set	Validation set		
1	Parametric bumps ($Re_H = 13\,250\text{--}25\,200$)	Parametric bump ($Re_H = 27\,850$)	Periodic hill ($\alpha = 1.0$, $Re_H = 5600$)	Evaluation of cross-dataset predictive performance.
2	All two-dimensional flows except periodic hill ($\alpha = 1.5$, $Re_H = 5600$)	Periodic hill ($\alpha = 1.5$, $Re_H = 5600$)	Three-dimensional duct flow ($Re_H = 2000$)	Evaluation of cross-dataset and cross-dimension predictive performance.

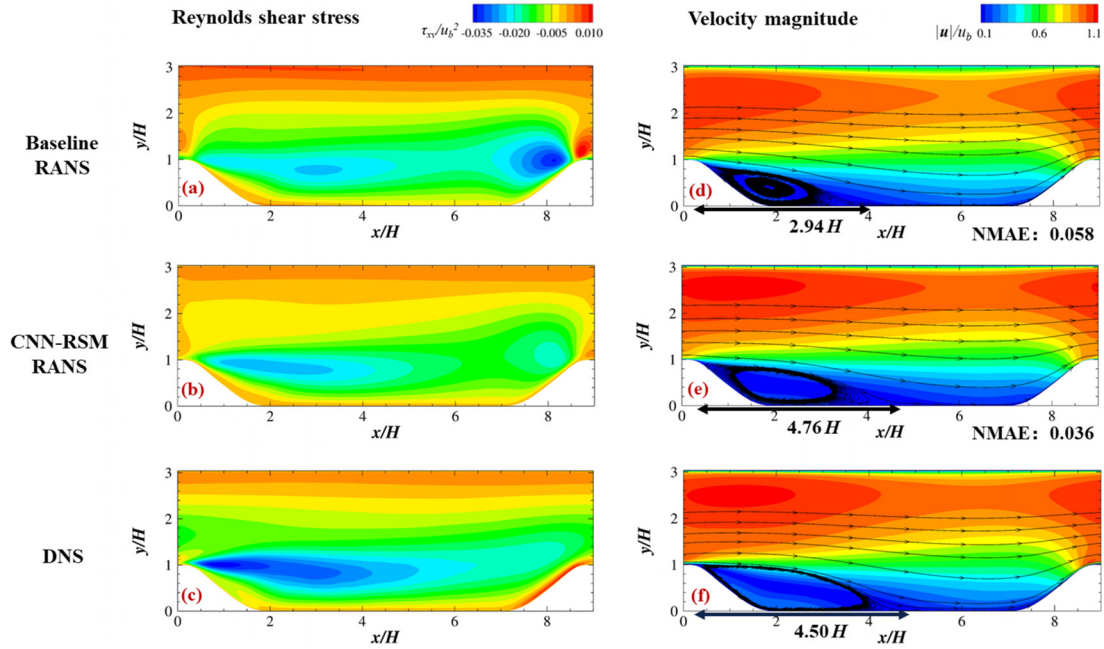


FIG. 3. Comparison of Reynolds shear stress and velocity magnitude distributions for the periodic hill case calculated by baseline RANS using the $k-\epsilon$ model, CNN-RSM RANS, and DNS: (a)–(c) Reynolds shear stress, normalized by the square of the bulk velocity u_b^2 ($u_b = 0.028$ m/s), and (d)–(f) velocity magnitude, normalized by u_b , with the NMAEs and reattachment lengths displayed at the bottom of each subfigure.

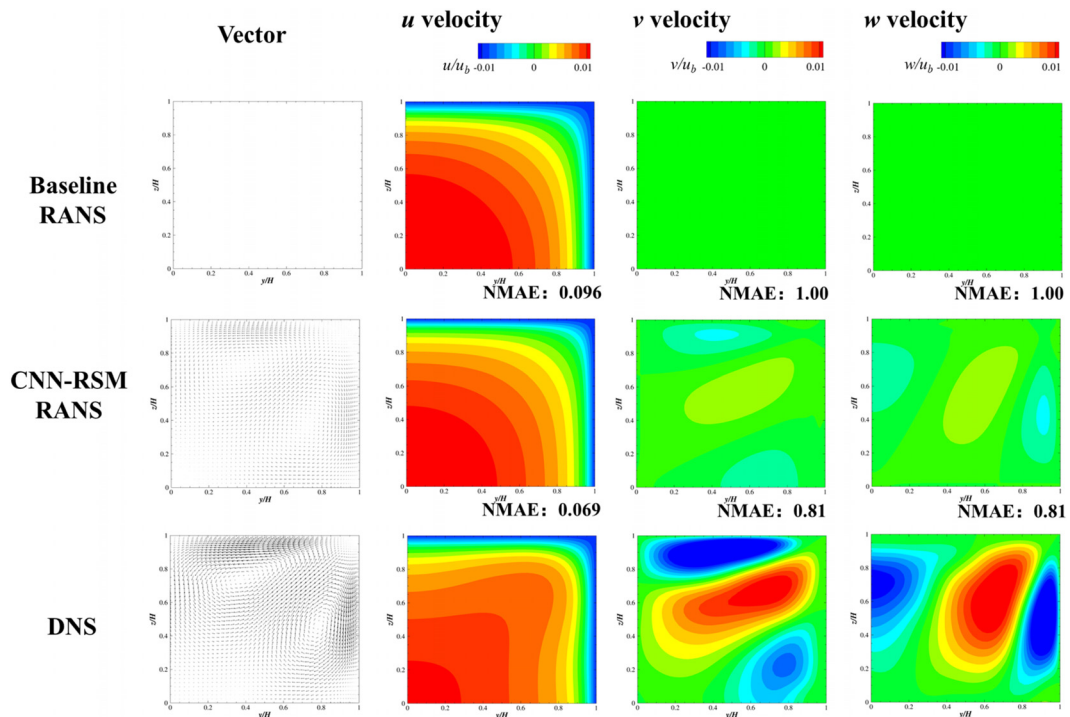


FIG. 4. Comparison of the velocities in the duct cross section, normalized by the bulk velocity ($u_b = 0.482$ m/s). From left to right, the first column displays the velocity vectors, and the other columns display the velocity components u , v , and w . The rows display the results of RANS simulation using the $k-\epsilon$ model, CNN-RSM RANS and DNS. The NMAEs are provided in the lower right corner of each subfigure.

indicating a closer match to the DNS result. Moreover, CNN-RSM RANS accurately predicted the general profiles and signs of the velocity components v and w , despite underestimating their magnitudes, whereas the baseline RANS predicted both velocity components as zero. Overall, although the CNN model was trained on a two-dimensional dataset, where the Reynolds stress components τ_{xz} and τ_{yz} were zero, it successfully predicted these components and reproduced the secondary flow in the duct. This demonstrates that the proposed CNN-RSM RANS can be applied across dimensions. This ability arises because the CNN model predicts the coefficients of the tensor bases instead of directly estimating the Reynolds stress components.

IV. APPLICATIONS TO PROBLEMS WITH MORE COMPLEX GEOMETRIES

To further assess the applicability of the proposed framework, this study conducted three case studies with available wind tunnel data for comparison. In addition to the wind tunnel experimental result, LES and RANS simulations were conducted for comparison and evaluation of the performance. The LES results were employed as a benchmark representing high-accuracy simulation. In addition, as the CNN-RSM model was trained based on the standard $k-\epsilon$ model, the standard $k-\epsilon$ was employed for the RANS simulation to ensure a fair comparison. Note that due to the lack of high-fidelity Reynolds stress data, only the velocity fields calculated by different methods were

compared qualitatively and quantitatively. The computing time was also compared for different methods. All the simulations were performed on a PC with a 24-core CPU (Intel core i9-13900K; 3.00 GHz; 64 GB memory). The CNN model employed in this section was trained on all the datasets described in Sec. III A to best ensure the performance of the model. The validation set selected was the three-dimensional duct flow ($Re_H = 2000$), as the trained model must have sufficient accuracy for 3D applications.

A. Case 1: Flow around an isolated building

Case 1 is the wind environment around a 1:1:2 (width: depth: height) shape isolated building previously studied by Yan *et al.*,⁶⁷ as illustrated in Fig. 5(a). In the wind tunnel experiment, the height of the target building was 0.16 m, and the mean inflow velocity followed a vertical profile power law with an exponent of 0.27, measuring 4.491 m/s at the building height. The characteristic Reynolds number based on building height and the mean inflow velocity at building height was 51 100. There were 186 measurement points located on the vertical plane at $y = 0$ m and the horizontal planes at $z = 0.125$ m and $z = 1.25$ m, as shown in Figs. 5(b) and 5(c). The computational domain size was adopted from the Architectural Institute of Japan (AIJ) guidelines⁶⁸ and was $15.5H \times 10.5H \times 6H$, where H is the height of the building. The resulting blockage ratio for this computational domain was 0.79%, below the recommended limit of 3%.⁶⁹ For the RANS simulation, the standard $k-\epsilon$ model was used. The velocity and

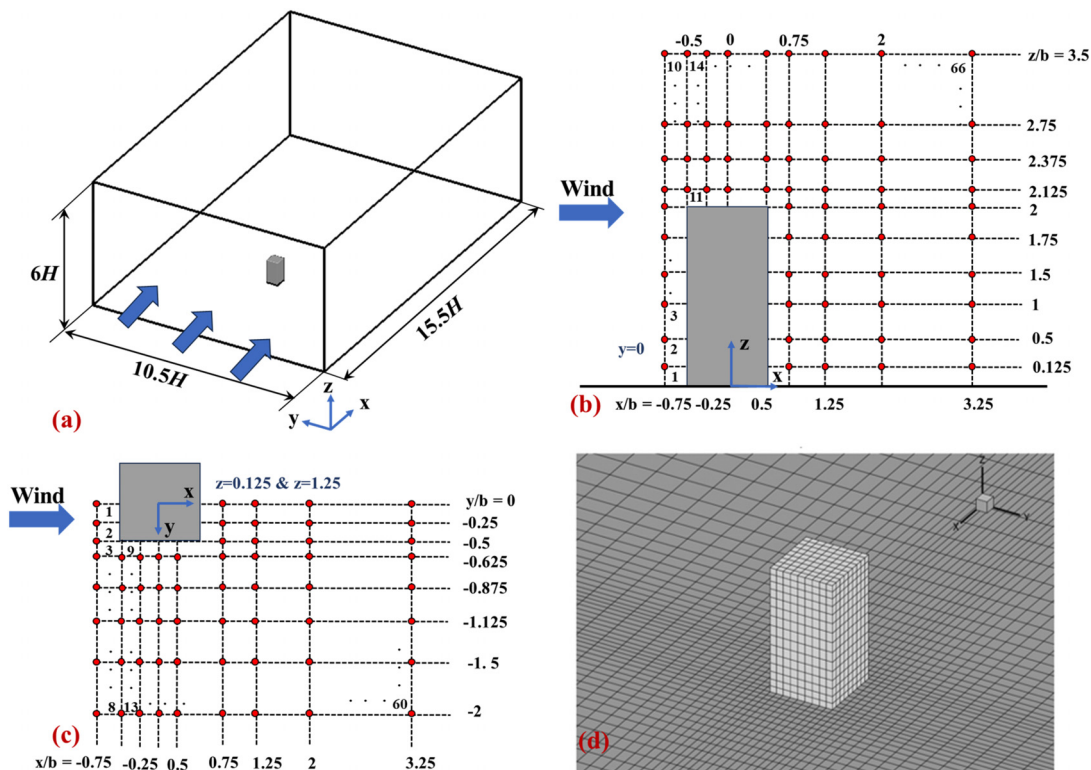


FIG. 5. (a) Computational domain of the isolated building model, (b) measurement points in vertical plane ($y = 0$ m), (c) measurement points in the horizontal plane ($z = 0.125$ m and $z = 1.25$ m), and (d) grid discretization.

TKE inflow boundaries were obtained by interpolating the profile from the experiment⁶⁷ and the turbulent dissipation rate ε was calculated by the following equation:

$$\varepsilon = C_\mu^{1/2} k \frac{\partial u}{\partial z}, \quad (11)$$

where C_μ is a constant of 0.09, and z is the height. The building surface was set as a non-slip wall; the outlet and lateral boundaries were set as zero-gradient boundaries; the ground surface was set to a rough wall with a roughness length of 1.8×10^{-4} m; and the top boundary was set as a slip wall. The second-order upwind scheme was applied for all the advection terms. A uniform structured grid was used to discretize the computational domain, as shown in Fig. 5(d), and a grid-independence test was performed on grids with 0.22×10^6 , 0.62×10^6 , and 2.10×10^6 cells. The results showed that the grid with 0.62×10^6 cells was sufficiently fine.

For the LES simulation, the Smagorinsky model⁷⁰ with $C_s = 0.14$ was used as the subgrid scale (SGS) model to calculate turbulence. The inlet velocity fluctuations were generated by the narrowband synthesis random flow generator (NSRFG) method,⁷¹ and the other boundary conditions were consistent with the RANS settings. The linear-upwind stabilized transport (LUST) scheme was applied for advection terms. The scheme selected for time integration was the second-order backward difference scheme, and the discretization time step was set as 0.005 s, resulting in a Courant–Friedrichs–Lewy (CFL) number of 1.90. The sampling duration was 10 s, at which point the time-averaged velocities at each measurement point had reached stability.⁷²

Note that the LES computation was initialized with the solution from the RANS simulation. To mitigate the influence of the initial state, the LES was run for a duration equivalent to five flow-through domains (FTD) before the collection of sampling data.⁷³ Here, the FTD is defined as L/u_H , where L is the domain length and u_H is the mean inlet velocity at H . This duration corresponded to 2.76 s, so the total simulation time for LES was 12.76 s. A grid-independence test was conducted on grids with 0.59×10^6 , 1.19×10^6 , and 2.40×10^6 cells, and the grid containing 1.19×10^6 cells was selected for LES simulation.

Figure 6 shows the velocity contours, streamlines, and reattachment lengths on the roof and behind the building. For the baseline RANS model, the reattachment length behind the building was significantly higher than the experimental value, and no reverse flow was observed on the roof surface. In contrast, the CNN-RSM RANS model successfully reproduced the flow reattachment on the roof surface with a length of $0.30 H$, greater than the experimental value of $0.26 H$. Additionally, although the reattachment length behind the building was still greater than the experimental value, it was effectively shortened compared to the baseline RANS model. The LES model, however, predicted that the reattachment length behind the building closely matched the experimental value, but that the length on the roof surface was greater than both the experimental result and the result produced by the CNN-RSM RANS. Overall, the CNN-RSM RANS model outperformed the baseline RANS in predicting the reattachment both on the roof and behind the building, particularly delivering the most accurate result for the roof reattachment, in close alignment with experimental data.

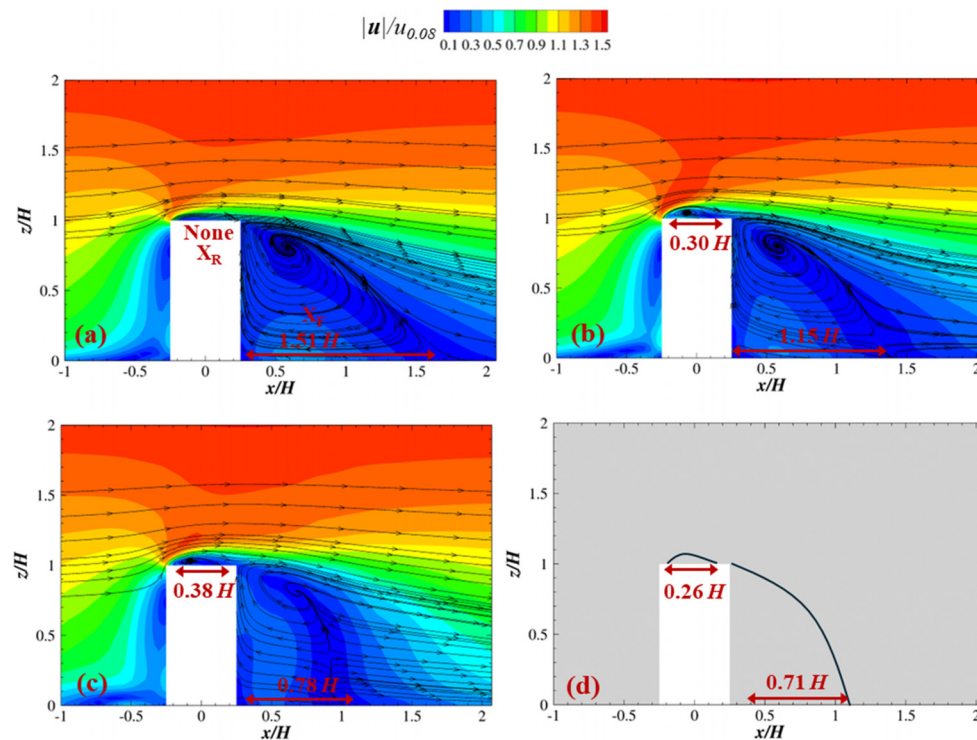


FIG. 6. Comparison of velocity magnitude contours with streamlines and reattachment length of the isolated building case: (a) baseline RANS, (b) CNN-RSM RANS, (c) LES, and (d) experiment. All the velocity magnitudes were normalized by the mean inlet velocity at $z = 0.08$ m ($u_{0.08} = 3.824$ m/s).

Figure 7 compares the velocity profiles at different locations as calculated by different methods. The flow field generated by the CNN-RSM RANS model matched the experimental data well across all planes, exhibiting a smooth and consistent distribution without irregularities. Specifically, on the vertical plane at $y = 0$ m, the CNN-RSM RANS results align well with experimental data. This method accurately predicted the u velocity near the top of the building, performing even better than the LES model. Behind the building, CNN-RSM RANS produced a velocity distribution similar to that of the baseline RANS, and it agreed better with experimental values near the ground. On the horizontal plane at $z = 0.125$ m, the LES model performed adequately behind the building but poorly in the side wind zone, especially in areas farther away from the building. In contrast, the CNN-RSM RANS results were closest to the experimental results in these regions among the different numerical prediction methods. On the horizontal plane at $z = 1.25$ m, it can be seen that CNN-RSM RANS performed poorly in predicting velocities in the region behind the building, and thus not as effectively as in the lower plane. Nevertheless, the CNN-RSM RANS model still demonstrated good accuracy in predicting the wind distribution in most other surrounding areas.

For rigorous quantification of the prediction accuracy, the root mean square error (RMSE) and correlation coefficient were introduced to quantify the prediction error and the degree of correlation between the predicted velocities and the wind tunnel data, and are defined as

$$R = \frac{\text{Cov}(u_{\text{ref}}, u_{\text{pred}})}{\sqrt{\text{Var}(u_{\text{ref}})\text{Var}(u_{\text{pred}})}}, \quad (12)$$

$$\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^m (u_{\text{ref}} - u_{\text{pred}})^2}, \quad (13)$$

where R is the correlation coefficient, u_{ref} is the wind tunnel data, u_{pred} is the velocity calculated by different models, $\text{Cov}(\cdot)$ is the covariance, $\text{Var}(\cdot)$ is the variance, and m is the total number of measurement points. Table III provides quantitative comparisons of the RMSE and correlation coefficient of velocity magnitude as well as computing time. The CNN-RSM RANS model significantly outperformed the baseline RANS model in terms of RMSE, showing an improvement of 18.2%. This improvement also slightly exceeded the performance of the LES model, which had a RMSE of 0.46. Moreover, in terms of the

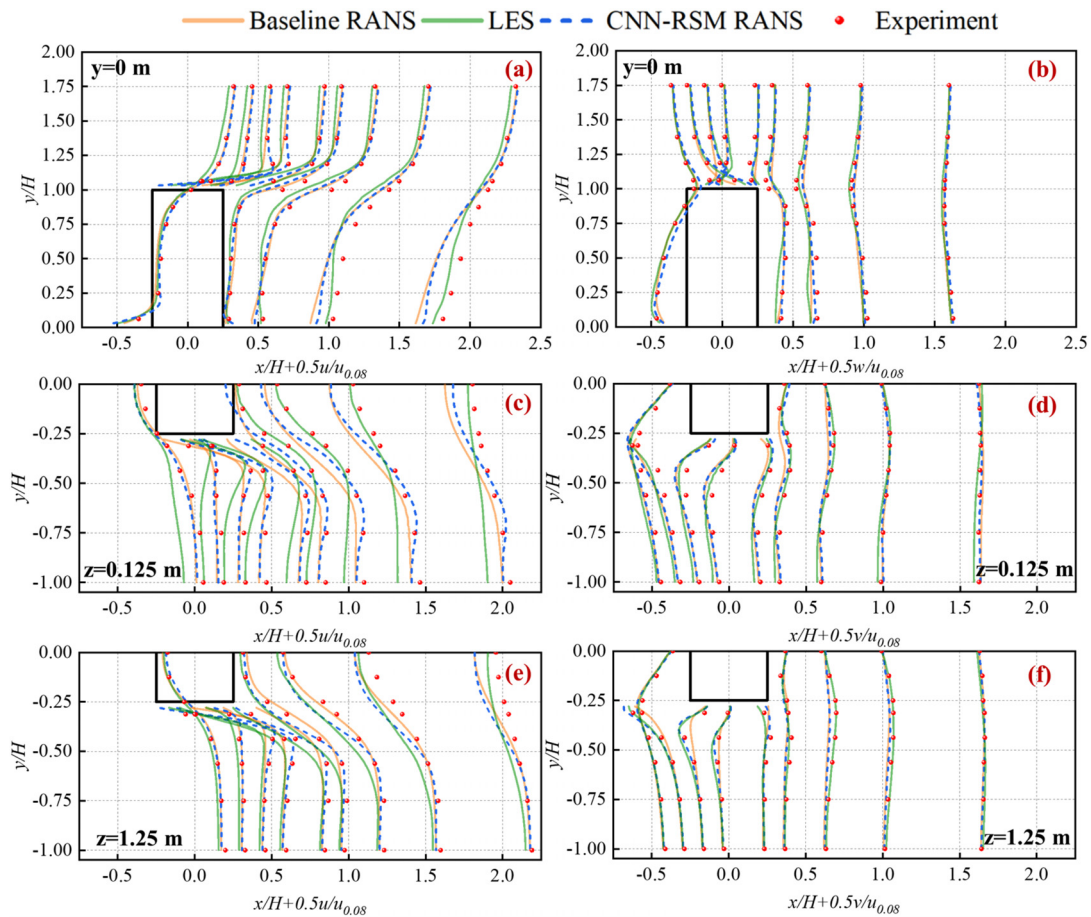


FIG. 7. Comparison of velocity component profiles (u , v , and w) for the single building case, normalized by the mean inlet velocity at $z = 0.08$ m ($u_{0.08} = 3.824$ m/s). The profiles were measured in three planes, including the vertical plane at $y = 0$ m and the horizontal planes at $z = 0.125$ m and $z = 1.25$ m: (a) u in $y = 0$ m plane, (b) v in $y = 0$ m plane, (c) u in $z = 0.125$ m plane, (d) v in $z = 0.125$ m plane, (e) u in $z = 1.25$ m plane, and (f) v in $z = 1.25$ m plane.

TABLE III. Quantitative analysis for the isolated building case: RMSE, correlation coefficient, and computing time.

	Baseline RANS	LES	CNN-RSM RANS
RMSE	0.55	0.46	0.45
Correlation coefficient R	0.95	0.97	0.97
Computing time (CPU hours)	2.04	33.61	3.60

correlation coefficient, CNN-RSM RANS achieved a coefficient of 0.97, equaling the performance of the LES model. This result also outperformed the correlation coefficient of 0.95 for the baseline RANS model, further demonstrating the superiority of the CNN-RSM RANS model in terms of predictive effectiveness. Furthermore, in terms of computing efficiency, compared to the more costly LES model, the CNN-RSM RANS model required only 1.76 times the computing time required by the baseline RANS model, which is affordable for many practical applications. Note that the computing time for the CNN-RSM RANS listed in the table represents the total time required during the application phase. Even when including the computing time spent on model training (2.4 h), the total time required by the CNN-RSM RANS model was still significantly lower than that required for the LES. Overall, the results demonstrated that the CNN-RSM RANS model provided a significant advantage over the LES model, achieving a substantial reduction in resource utilization while maintaining a high level of accuracy.

B. Case 2: Flow around a building array

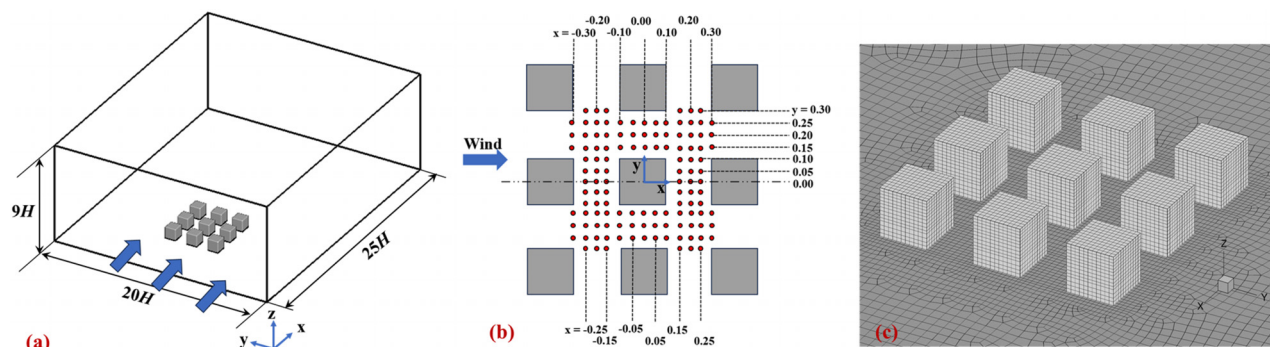
Case 2 is the wind environment around an idealized building array with nine building blocks studied by Nonomura *et al.*,⁷⁴ as illustrated in Fig. 8(a). The building blocks were cuboids, with each side measuring 0.2 m, and the mean velocity of the inflow at building height was 3.654 m/s, resulting in a characteristic Reynolds number of 52 000. There were 120 measurement points on the horizontal plane at $z = 0.02$ m in the wind tunnel experiment, as shown in Fig. 8(b). The computational domain size was $25H \times 20H \times 9H$, resulting in a blockage ratio of 1.67%. The turbulence model and boundary conditions were configured using the same method as in case 1, with the ground surface roughness length set to 4.5×10^{-4} m. The first-order upwind

scheme was applied for all the advection terms. A non-uniform grid was employed to discretize the computational domain, as shown in Fig. 8(c). A grid-independence test was performed on grids with 0.35×10^6 , 0.59×10^6 , and 0.97×10^6 cells, and the result showed that the grid with 0.59×10^6 cells was sufficiently fine to capture the turbulence.

For the LES simulation, the selected SGS model was the Smagorinsky model with $C_s = 0.10$. The boundary conditions and numerical schemes were configured identically to those in case 1. The discretization time step was set to 0.005 s, resulting in a CFL number of 1.12. The sampling time was set at 17 s, and the total simulation time was 23.84 s. A grid-independence test was performed on grids with 0.95×10^6 , 2.03×10^6 , and 4.05×10^6 cells, and the grid with 2.03×10^6 cells was selected for LES simulation.

Figure 9 compares the normalized velocity magnitude calculated by baseline RANS, LES, and CNN-RSM RANS for this case, along with experimental values. In general, the velocity magnitude obtained by LES was closest to the true value, while the results of the baseline RANS and CNN-RSM RANS models were found to be lower than the experimental values. For points located in the building canyons parallel to the inflow as in Fig. 9(b), the CNN-RSM RANS demonstrated greater accuracy than the baseline RANS, as the flow characteristics were simple and similar to those in the training dataset. However, in the complex flow regions behind buildings as in Fig. 9(c), the CNN-RSM RANS prediction fell short. This underperformance was likely due to the strong recirculation in these areas, which was not represented in the training data. These results suggest that CNN-RSM RANS performs well in regions where the flow dynamics are similar to those of the training data. However, it tends to underperform relative to the baseline RANS in scenarios involving flow patterns that are not represented in the training dataset. Therefore, future efforts may focus on expanding the training dataset to include more flows related to outdoor wind distribution.

Table IV provides quantitative comparisons of the RMSE and correlation coefficient of velocity magnitude as well as computing time. In general, the CNN-RSM RANS model showed improvement over the baseline RANS. While the improvement in RMSE was modest, the correlation coefficient improved appreciably. Moreover, the CNN-RSM RANS only required 1.62 times the time required by the baseline RANS, which is significantly less than the 38.96 times the time required by LES. Even when accounting for the training time, the overall computational time required by the CNN-RSM RANS

**FIG. 8.** (a) Computational domain of the building array model, (b) measurement points in the horizontal plane ($z = 0.02$ m), and (c) grid discretization.

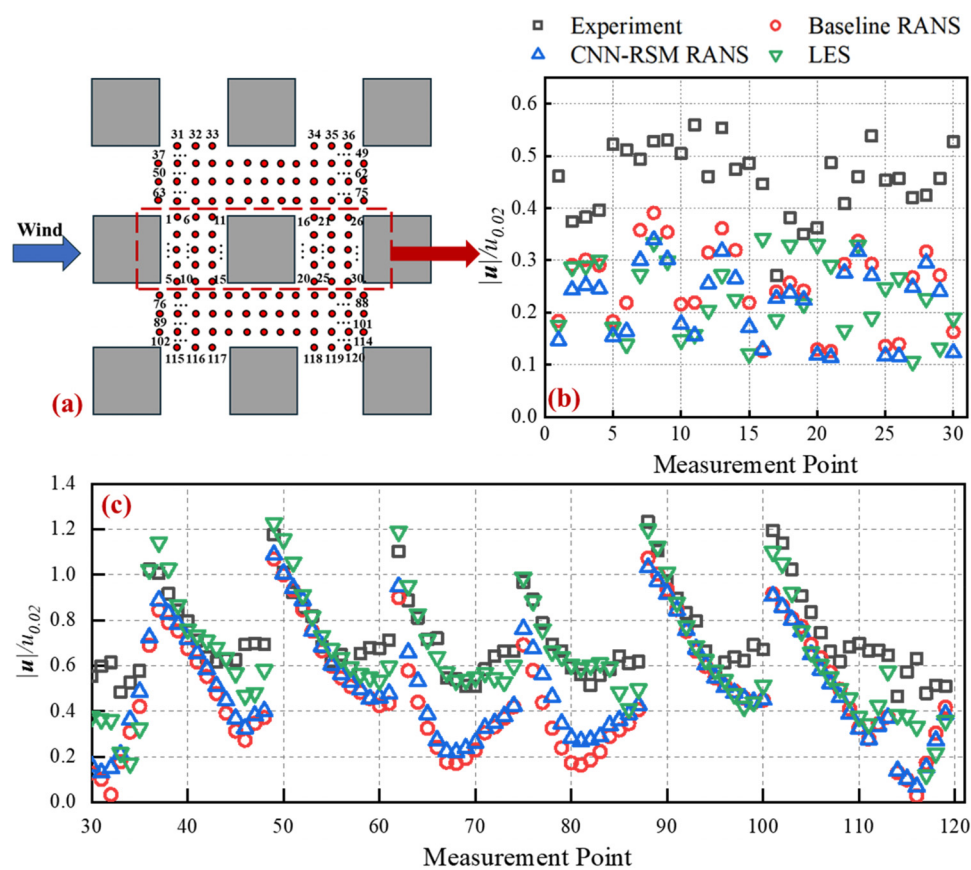


FIG. 9. Comparison of velocity magnitude at each measurement point of the building array case, normalized by the mean inlet velocity at $z = 0.02$ m ($u_{0.02} = 2.434$ m/s): (a) numbering of measurement points, (b) normalized velocity magnitude for points 1–30 behind buildings, and (c) normalized velocity magnitude for the remaining points.

remained significantly lower than the extensive computational demands of the LES, making it a more time-effective solution for practical applications.

C. Case 3: Flow around building complexes in Niigata

Case 3 is the wind environment around building complexes with simple building shapes in Niigata. As shown in Fig. 10(a), in the building complex, there was a high-rise building (building A) with a height of 60 m and two mid-rise buildings (buildings B and C), each with a height of 18 m. The wind tunnel data were collected in a 1:250 scale model by Tominaga *et al.*⁷⁵ The mean velocity of the approaching flow

at the height of the scaled building A was 5.343 m/s, resulting in a characteristic Reynolds number of 91 200. The measurement points were at a height of 0.008 m from the ground in the model scale, as shown in Fig. 10(b). The computational domain is shown in Fig. 10(c) and resulted in a blockage ratio of 1.39%. The turbulence model, numerical schemes, and boundary conditions were configured using the same method as in case 1, with the roughness length of the ground surface set to 9.6×10^{-5} m. A non-uniform grid was employed for the discretization of the computational domain, as shown in Fig. 10(d). A grid-independence test was performed on grids with 4.24×10^6 , 6.63×10^6 , and 9.90×10^6 cells. The result showed that the grid with 6.63×10^6 mesh was sufficiently fine for the calculation of turbulence.

For the LES simulation, the setup for boundary conditions, SGS model, and numerical schemes was identical to those in case 1. The discretization time step was 0.002 s, resulting in a CFL number of 0.93. The sampling duration was 12 s. Before sampling, the LES calculation was performed for 4.88 s, equivalent to 5 FTD. A grid-independence test was performed on grids with 6.63×10^6 , 9.80×10^6 , and 12.93×10^6 cells, and the grid with 9.80×10^6 cells was selected for LES simulation.

Figure 11 displays the velocity contours around the measurement points, along with a pointwise comparison of the normalized velocity magnitudes calculated by the baseline RANS, LES, and CNN-RSM

TABLE IV. Quantitative analysis for RMSE, correlation coefficient, and computing time of the building array case.

	Baseline RANS	LES	CNN-RSM RANS
RMSE	0.26	0.18	0.25
Correlation coefficient	0.85	0.96	0.89
Computing time (CPU hours)	2.67	104.03	4.34

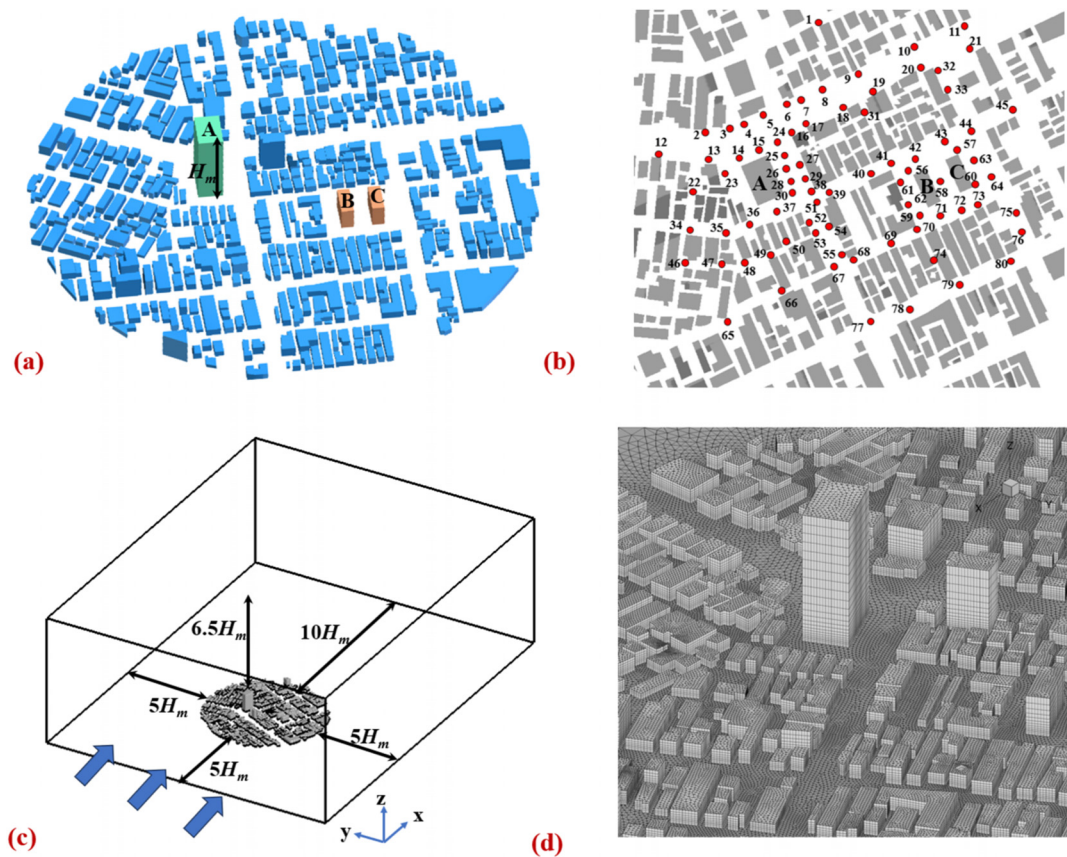


FIG. 10. (a) Configuration of the building complexes, (b) measurement points in horizontal plane ($z = 0.008$ m), (c) computational domain, and (d) grid discretization.

RANS models. Generally, the velocity contours obtained by the RANS and CNN-RSM RANS were similar, but the velocity magnitude predicted by CNN-RSM RANS was slightly lower than the RANS result in several areas, such as the region on the left of building A. Compared to these two methods, the LES method showed higher velocities around building A, and lower velocities around buildings B and C. As shown in Fig. 11(c), the velocities predicted by both the baseline RANS and CNN-RSM RANS were lower than the wind tunnel experiment result. Additionally, at most of the measurement points, CNN-RSM RANS predicted slightly lower velocities than the baseline RANS.

Table V summarizes the RMSE, correlation coefficient and computing time for the baseline RANS, LES, and CNN-RSM RANS. According to the results, LES demonstrated a slight superiority in terms of RMSE and correlation coefficient compared to the other two methods, but it was considerably more time-consuming. In addition, the correlation coefficients and RMSEs of the velocity magnitude predicted by the CNN-RSM RANS model were slightly worse than the RANS results. This can be attributed to the complexity of the geometry and grid system employed in this test. Overall, these results indicate that CNN-RSM RANS can generate continuous velocity fields even in complex geometry. Although the predicted velocity field is not as accurate as that of RANS, this is still an encouraging result as it demonstrates the high robustness of the proposed framework. Furthermore,

for such complex urban configurations, the computational resource consumption of the CNN-RSM RANS is 36% higher than that of the baseline RANS, and the time required for model training is negligible compared to the simulation time.

V. DISCUSSION AND LIMITATIONS

The development of this machine learning framework prioritized generalization capability and robustness, potentially compromising the accuracy of *a posteriori* results. This is largely due to the fact that the conservative strategy employed by the calculation algorithm was designed to balance accuracy with numerical stability. While the stability of this approach was validated in complex urban applications, further investigations should focus on improving accuracy without compromising stability. Moreover, since there are no available physical constraints for the prediction of the TKE k , the discrepancy $\Delta \ln k$ was confined within the lower and upper bounds of the training data to avoid unjustified extrapolation. In our experience, this has occasionally resulted in nonphysical predictions, especially under highly complex flow conditions. In such instances, a viable option is to use the k calculated by RANS instead of the modified k to ensure that the Reynolds stress distribution aligns with physical norms. Therefore, it is essential to continue research on the integration of appropriate constraints and enhancements in the prediction capabilities of k .

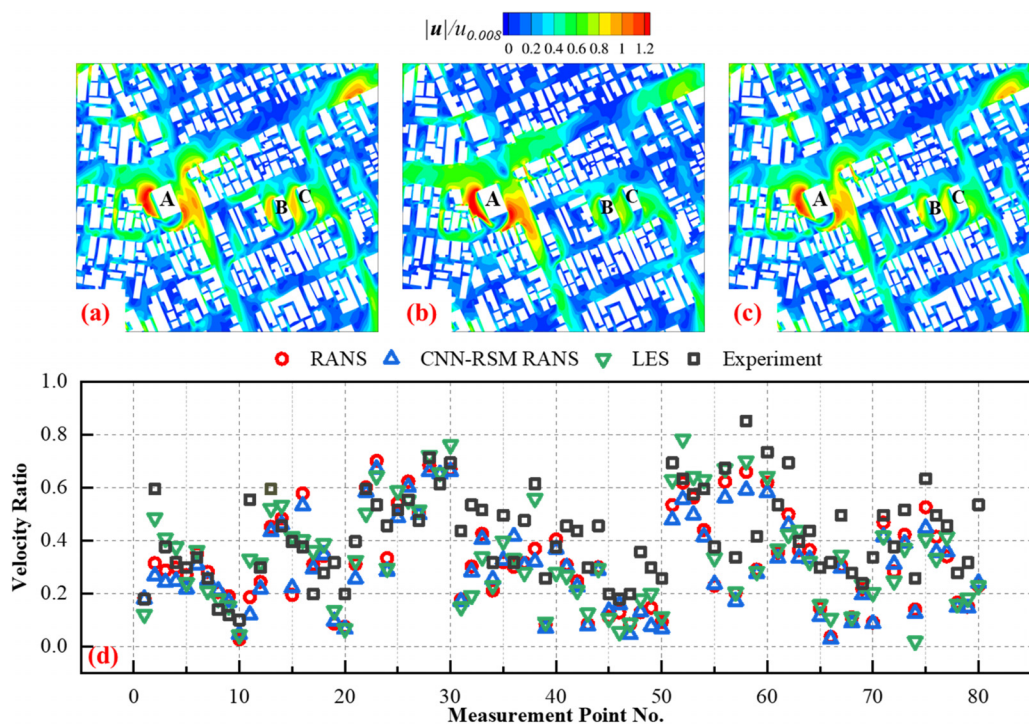


FIG. 11. Comparison of the velocity magnitude, normalized by the mean inlet velocity at $z = 0.008$ m ($u_{0.008} = 2.964$ m/s): (a) contour of the baseline RANS, (b) contour of the LES, (c) contour of the CNN-RSM RANS, and (d) pointwise comparison at each measurement point between the baseline RANS, LES, and CNN-RSM RANS.

The present study employed a limited number of high-fidelity datasets that lacked relevance to typical outdoor flows. The extension of the dataset is expected to enhance the predictive performance and generalizability of the model. However, as the focus of this research was on the construction of the CNN-RSM RANS turbulence modeling framework, the dataset that had been widely employed and validated in related literature was utilized. This utilization facilitated a systematic comparison of the performance of the model with other existing models during the development process, enabling the evaluation of the advantages and potential areas of the model for improvement. Future research will expand the dataset and explore the impact of different types of high-fidelity data on model performance, thereby further enhancing the practicality and generalizability of the model. Furthermore, it is important to note that the present framework was designed for steady RANS computations. Although it exhibits excellent performance, it can only calculate time-averaged flow variables. For

TABLE V. Quantitative analysis for the building complex case: RMSE, correlation coefficient, and computing time of the building complex case.

	Baseline RANS	LES	CNN-RSM RANS
RMSE	0.15	0.14	0.16
Correlation coefficient	0.80	0.82	0.77
Computing time (CPU hours)	551.81	3949.32	751.82

some applications, such as pollutant dispersion and thermal comfort, transient flow data are preferred. Therefore, extending the present framework to unsteady RANS computations would be beneficial. Future research should explore this direction to fully leverage the capabilities of the present framework.

VI. CONCLUSION

This study aimed to develop a machine learning model that incorporated the spatial characteristics of the flow field to provide continuous and accurate outdoor wind distributions. A CNN model was developed to capture spatial correlations and learn the relationship between inputs from the RANS flow field and the high-fidelity Reynolds stress. The trained model was then integrated into the CFD solver to compute the flow variables of interest. The generalization capability and practicality of the proposed framework were demonstrated through two tests and three outdoor wind environment case studies. Within the scope of this study, the following conclusions can be drawn:

- The CNN model provided continuous and smooth Reynolds stress predictions and demonstrated the ability to predict across datasets and dimensions.
- The CNN-RSM RANS model demonstrated satisfactory practicality by providing a more accurate prediction of the velocity field around an isolated building than the baseline RANS model.
- For more complex geometries, the CNN-RSM RANS model performed well in regions with flow conditions similar to those in

the training dataset, but the prediction accuracy needs to be improved in regions where the flow patterns have not been well covered in the training process.

- Even in highly complicated applications, the CNN-RSM RANS model provided a continuous and smooth velocity field distribution, which demonstrates the robustness of the proposed turbulence modeling framework.

ACKNOWLEDGMENTS

This work was supported by the Theme-based Research Scheme (Grant No. 22-504/21-R) from the Research Grants Council of Hong Kong SAR, China.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Rui Zhao: Investigation (lead); Methodology (lead); Writing – original draft (lead). **Siyang Zhong:** Writing – review & editing (equal); Supervision (supporting). **Ruoyu You:** Conceptualization (lead); Funding acquisition (lead); Supervision (lead); Writing – review & editing (equal).

DATA AVAILABILITY

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

APPENDIX: REYNOLDS STRESS COMPONENTS OF THE DUCT FLOW CASE

Comparison of Reynolds stress components of the duct flow case, normalized by the square of the bulk velocity u_b^2 ($u_b = 0.482$ m/s). (Fig. 12).

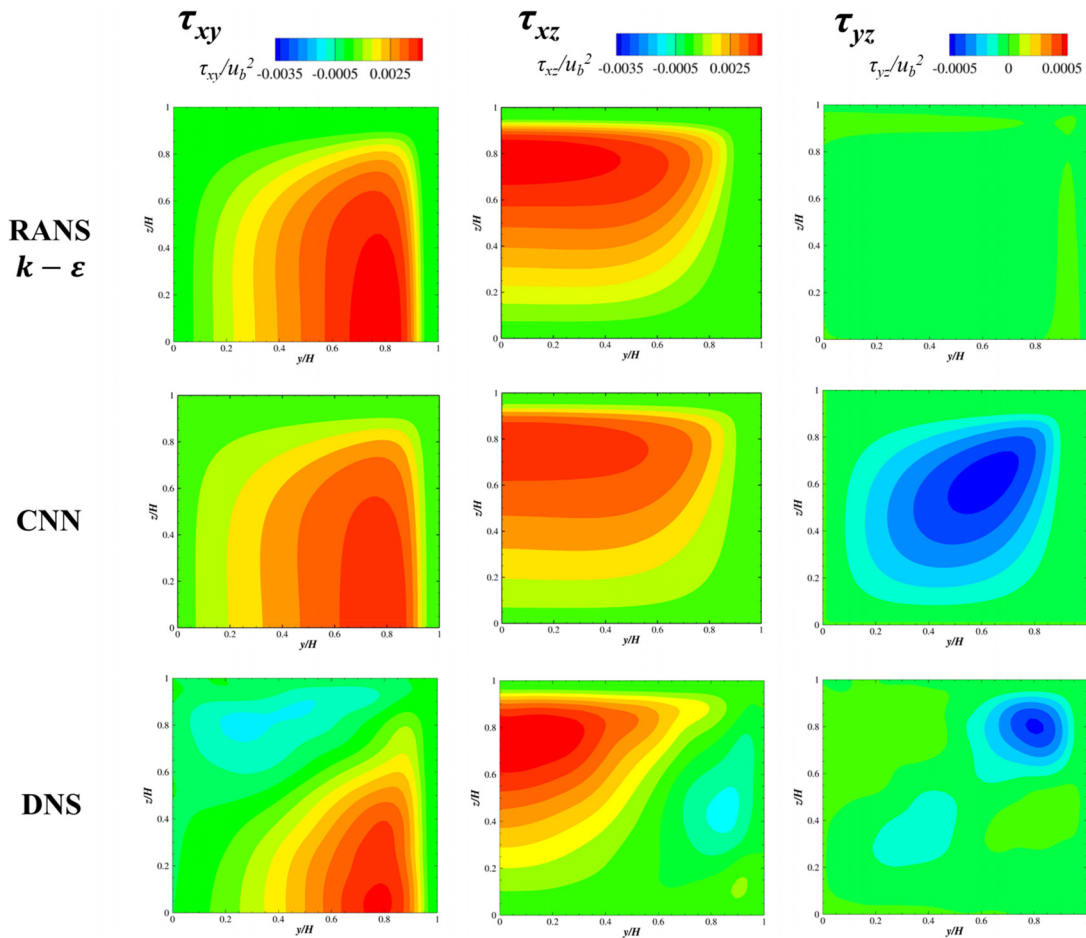


FIG. 12. Comparison of Reynolds stress components of the duct flow case, normalized by the square of the bulk velocity u_b^2 ($u_b = 0.482$ m/s). The columns display the Reynolds stress components of the τ_{xy} , τ_{xz} , and τ_{yz} , while the rows display the results from the RANS $k-\epsilon$ model, CNN, and DNS.

REFERENCES

- ¹T. S. Church, D. M. Thomas, C. Tudor-Locke, P. T. Katzmarzyk, C. P. Earnest, R. Q. Rodarte, C. K. Martin, S. N. Blair, and C. Bouchard, "Trends over 5 decades in U.S. occupation-related physical activity and their associations with obesity," *PLoS One* **6**(5), e19657 (2011).
- ²K. M. M. Beyer, A. Szabo, K. Hoormann, and M. Stolley, "Time spent outdoors, activity levels, and chronic disease among American adults," *J. Behav. Med.* **41**(4), 494–503 (2018).
- ³D. Zhu, G. Liu, Z. Lv, S. Wen, S. Bi, and W. Wang, "Inverse associations of outdoor activity and vitamin D intake with the risk of Parkinson's disease," *J. Zhejiang Univ. Sci. B* **15**(10), 923–927 (2014).
- ⁴M. Owens and H. L. I. Bunce, "The potential for outdoor nature-based interventions in the treatment and prevention of depression," *Front. Psychol.* **13**, 740210 (2022).
- ⁵M. A. Eusuf, M. A. Mohit, M. M. R. S. Eusuf, and M. Ibrahim, "Impact of outdoor environment to the quality of life," *Procedia – Soc. Behav. Sci.* **153**, 639–654 (2014).
- ⁶Y. Du, C. M. Mak, K. Kwok, K.-T. Tse, T. Lee, Z. Ai, J. Liu, and J. Niu, "New criteria for assessing low wind environment at pedestrian level in Hong Kong," *Build. Environ.* **123**, 23–36 (2017).
- ⁷K.-N. Kang, D. Song, and S. Schiavon, "Correlations in thermal comfort and natural wind," *J. Therm. Biol.* **38**(7), 419–426 (2013).
- ⁸Y. Zhou, Y. An, W. Huang, C. Chen, and R. You, "A combined deep learning and physical modeling method for estimating air pollutants' source location and emission profile in street canyons," *Build. Environ.* **219**, 109246 (2022).
- ⁹C. García-Sánchez, J. van Beeck, and C. Górlé, "Predictive large eddy simulations for urban flows: Challenges and opportunities," *Build. Environ.* **139**, 146–156 (2018).
- ¹⁰Y. Tominaga and T. Stathopoulos, "CFD modeling of pollution dispersion in a street canyon: Comparison between LES and RANS," *J. Wind Eng. Ind. Aerodyn.* **99**(4), 340–348 (2011).
- ¹¹G. Vita, S. Salvadori, D. A. Misul, and H. Hemida, "Effects of inflow condition on RANS and LES predictions of the flow around a high-rise building," *Fluids* **5**(4), 233 (2020).
- ¹²G. Lamberti and C. Górlé, "Sensitivity of LES predictions of wind loading on a high-rise building to the inflow boundary condition," *J. Wind Eng. Ind. Aerodyn.* **206**, 104370 (2020).
- ¹³A. A. Adedoyin, D. K. Walters, and S. Bhushan, "Investigation of turbulence model and numerical scheme combinations for practical finite-volume large eddy simulations," *Eng. Appl. Comput. Fluid Mech.* **9**(1), 324–342 (2015).
- ¹⁴G. R. Tabor and M. H. Baba-Ahmadi, "Inlet conditions for large eddy simulation: A review," *Comput. Fluids* **39**(4), 553–567 (2010).
- ¹⁵X. Wu, "Inflow turbulence generation methods," *Annu. Rev. Fluid Mech.* **49**, 23–49 (2017).
- ¹⁶M. Shirzadi and Y. Tominaga, "CFD evaluation of mean and turbulent wind characteristics around a high-rise building affected by its surroundings," *Build. Environ.* **225**, 109637 (2022).
- ¹⁷S. M. Salim, R. Buccolieri, A. Chan, and S. Di Sabatino, "Numerical simulation of atmospheric pollutant dispersion in an urban street canyon: Comparison between RANS and LES," *J. Wind Eng. Ind. Aerodyn.* **99**(2), 103–113 (2011).
- ¹⁸J. Liu, M. Heidarnejad, G. Pitchurov, L. Zhang, and J. Srebric, "An extensive comparison of modified zero-equation, standard k- ϵ , and LES models in predicting urban airflow," *Sustainable Cities Soc.* **40**, 28–43 (2018).
- ¹⁹J. Boussinesq, *Essai Sur la Theorie Des Eaux Courantes* (Imprimerie Nationale, 1877).
- ²⁰T. B. Gatski and C. L. Rumsey, "Linear and nonlinear eddy viscosity models," in *Closure Strategies for Turbulent and Transitional Flows*, edited by B. E. Launder and N. D. Sandham (Cambridge University Press, Cambridge, 2002), pp. 9–46.
- ²¹F. Toja-Silva, C. Peralta, O. Lopez-Garcia, J. Navarro, and I. Cruz, "Roof region dependent wind potential assessment with different RANS turbulence models," *J. Wind Eng. Ind. Aerodyn.* **142**, 258–271 (2015).
- ²²C. Lin, R. Ooka, H. Kikumoto, T. Sato, and M. Arai, "CFD simulations on high-buoyancy gas dispersion in the wake of an isolated cubic building using steady RANS model and LES," *Build. Environ.* **188**, 107478 (2021).
- ²³H. Xiao and P. Cinnella, "Quantification of model uncertainty in RANS simulations: A review," *Prog. Aerosp. Sci.* **108**, 1–31 (2019).
- ²⁴C. García-Sánchez, G. Van Tendeloo, and C. Górlé, "Quantifying inflow uncertainties in RANS simulations of urban pollutant dispersion," *Atmos. Environ.* **161**, 263–273 (2017).
- ²⁵T. Berk and B. Ganapathisubramani, "Effects of vortex-induced velocity on the development of a synthetic jet issuing into a turbulent boundary layer," *J. Fluid Mech.* **870**, 651–679 (2019).
- ²⁶W. Engblom, N. Georgiadis, and A. Khavaran, "Investigation of variable-diffusion turbulence model correction for round jets," AIAA Paper No. 2005-3085, 2012.
- ²⁷P. E. Smirnov and F. R. Menter, "Sensitization of the SST turbulence model to rotation and curvature by applying the Spalart–Shur correction term," *J. Turbomach.* **131**, 041010 (2009).
- ²⁸P. R. Spalart and M. Shur, "On the sensitization of turbulence models to rotation and curvature," *Aerosp. Sci. Technol.* **1**(5), 297–302 (1997).
- ²⁹R. Kumar and A. Dewan, "URANS computations with buoyancy corrected turbulence models for turbulent thermal plume," *Int. J. Heat Mass Transfer* **72**, 680–689 (2014).
- ³⁰C. Yan, H. Li, Y. Zhang, and H. Chen, "Data-driven turbulence modeling in separated flows considering physical mechanism analysis," *Int. J. Heat Fluid Flow* **96**, 109004 (2022).
- ³¹D. Ching, A. Banko, P. Milani, and J. Eaton, "Machine learning modeling for RANS turbulent kinetic energy transport in 3D separated flows," in 11th International Symposium on Turbulence and Shear Flow Phenomena, Southampton, UK (2019).
- ³²K. Duraisamy, Z. J. Zhang, and A. P. Singh, "New approaches in turbulence and transition modeling using data-driven techniques," AIAA Paper No. 2015-1284, 2015.
- ³³A. P. Singh and K. Duraisamy, "Using field inversion to quantify functional errors in turbulence closures," *Phys. Fluids* **28**(4), 045110 (2016).
- ³⁴P. Spalart and S. Allmaras, "A one-equation turbulence model for aerodynamic flows," AIAA Paper No. 1992-439, 1992.
- ³⁵J. Ling, A. Kurzawski, and J. Templeton, "Reynolds averaged turbulence modeling using deep neural networks with embedded invariance," *J. Fluid Mech.* **807**, 155–166 (2016).
- ³⁶C. Sotgiu, B. Weigand, K. Semmler, and P. Wellinger, "Towards a general data-driven explicit algebraic Reynolds stress prediction framework," *Int. J. Heat Fluid Flow* **79**, 108454 (2019).
- ³⁷P. M. Milani, J. Ling, and J. K. Eaton, "On the generality of tensor basis neural networks for turbulent scalar flux modeling," *Int. Commun. Heat Mass Transfer* **128**, 105626 (2021).
- ³⁸J.-X. Wang, J.-L. Wu, and H. Xiao, "Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data," *Phys. Rev. Fluids* **2**(3), 034603 (2017).
- ³⁹M. L. A. Kaandorp and R. P. Dwight, "Data-driven modeling of the Reynolds stress tensor using random forests with invariance," *Comput. Fluids* **202**, 104497 (2020).
- ⁴⁰J. P. Panda and H. V. Warrior, "Evaluation of machine learning algorithms for predictive Reynolds stress transport modeling," *Acta Mech. Sin.* **38**(4), 321544 (2022).
- ⁴¹X. Shan, X. Sun, W. Cao, W. Zhang, and Z. Xia, "Modeling Reynolds stress anisotropy invariants via machine learning," *Acta Mech. Sin.* **40**(6), 323629 (2024).
- ⁴²L. Zhao, Q. Zhou, M. Li, and Z. Wang, "Evaluating different CFD surrogate modelling approaches for fast and accurate indoor environment simulation," *J. Build. Eng.* **95**, 110221 (2024).
- ⁴³S. B. Pope, "A more general effective-viscosity hypothesis," *J. Fluid Mech.* **72**(2), 331–340 (1975).
- ⁴⁴Y. Yin, Z. Shen, Y. Zhang, H. Chen, and S. Fu, "An iterative data-driven turbulence modeling framework based on Reynolds stress representation," *Theor. Appl. Mech. Lett.* **12**(5), 100381 (2022).
- ⁴⁵S. Patankar, *Numerical Heat Transfer and Fluid Flow* (CRC Press, Boca Raton, 2018).
- ⁴⁶S. Wallin and A. V. Johansson, "An explicit algebraic Reynolds stress model for incompressible and compressible turbulent flows," *J. Fluid Mech.* **403**, 89–132 (2000).

- ⁴⁷W. Liu, J. Fang, S. Rolfo, C. Moulinec, and D. R. Emerson, "An iterative machine-learning framework for RANS turbulence modeling," *Int. J. Heat Fluid Flow* **90**, 108822 (2021).
- ⁴⁸D. B. Taulbee, J. R. Sonnenmeier, and K. M. Wall, "Stress relation for three-dimensional turbulent flows," *Phys. Fluids* **6**(3), 1399–1401 (1994).
- ⁴⁹J.-L. Wu, H. Xiao, and E. Paterson, "Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework," *Phys. Rev. Fluids* **3**(7), 074602 (2018).
- ⁵⁰W. Liu, J. Fang, S. Rolfo, C. Moulinec, and D. R. Emerson, "On the improvement of the extrapolation capability of an iterative machine-learning based RANS Framework," *Comput. Fluids* **256**, 105864 (2023).
- ⁵¹J. Ling and J. Templeton, "Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty," *Phys. Fluids* **27**(8), 085103 (2015).
- ⁵²S. Banerjee, R. Krah, F. Durst, and C. Zenger, "Presentation of anisotropy properties of turbulence, invariants versus eigenvalue approaches," *J. Turbul.* **8**, N32 (2007).
- ⁵³R. L. Thompson, L. E. B. Sampaio, F. A. V. de Bragança Alves, L. Thais, and G. Mompean, "A methodology to evaluate statistical errors in DNS data of plane channel flows," *Comput. Fluids* **130**, 1–7 (2016).
- ⁵⁴J. Wu, H. Xiao, R. Sun, and Q. Wang, "Reynolds-averaged Navier–Stokes equations with explicit data-driven Reynolds stress closure can be ill-conditioned," *J. Fluid Mech.* **869**, 553–586 (2019).
- ⁵⁵X. Guo, Z. Xia, and S. Chen, "Computing mean fields with known Reynolds stresses at steady state," *Theor. Appl. Mech. Lett.* **11**(3), 100244 (2021).
- ⁵⁶OpenFOAM: User Guide, Open source CFD: Documentation, OpenFOAM®, June 2022.
- ⁵⁷O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, edited by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi (Springer International Publishing, Cham, 2015), pp. 234–241.
- ⁵⁸I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv:1711.05101* (2019).
- ⁵⁹H. Xiao, J.-L. Wu, S. Laizet, and L. Duan, "Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations," *Comput. Fluids* **200**, 104431 (2020).
- ⁶⁰M. Marquillie, J.-P. Laval, and R. Dolganov, "Direct numerical simulation of a separated channel flow with a smooth profile," *J. Turbul.* **9**, N1 (2008).
- ⁶¹R. Matai and P. Durbin, "Large-eddy simulation of turbulent flow over a parametric set of bumps," *J. Fluid Mech.* **866**, 503–525 (2019).
- ⁶²H. Le, P. Moin, and J. Kim, "Direct numerical simulation of turbulent flow over a backward-facing step," *J. Fluid Mech.* **330**, 349–374 (1997).
- ⁶³J.-P. Laval and M. Marquillie, "Direct numerical simulations of converging–diverging channel flow," in *Progress in Wall Turbulence: Understanding and Modeling*, edited by M. Stanislas, J. Jimenez, and I. Marusic (Springer Netherlands, Dordrecht, 2011), pp. 203–209.
- ⁶⁴A. Pinelli, M. Uhlmann, A. Sekimoto, and G. Kawahara, "Reynolds number dependence of mean flow structure in square duct turbulence," *J. Fluid Mech.* **644**, 107–122 (2010).
- ⁶⁵R. McConkey, E. Yee, and F.-S. Lien, "A curated dataset for data-driven turbulence modeling," *Sci. Data* **8**(1), 255 (2021).
- ⁶⁶F. Pioch, J. H. Harmening, A. M. Müller, F.-J. Peitzmann, D. Schramm, and O. el Moctar, "Turbulence modeling for physics-informed neural networks: Comparison of different RANS models for the backward-facing step flow," *Fluids* **8**(2), 43 (2023).
- ⁶⁷M. Yan and H. Kazuki, "Turbulent measurements of the flow field around a high-rise building" (in Japanese), *Wind Eng.* **76**, 55–64 (1998).
- ⁶⁸Y. Tominaga, A. Mochida, R. Yoshie, H. Kataoka, T. Nozu, M. Yoshikawa, and T. Shirasawa, "AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings," *J. Wind Eng. Ind. Aerodyn.* **96**(10), 1749–1761 (2008).
- ⁶⁹J. Franke, A. Hellsten, K. H. Schlunzen, and B. Carissimo, "The COST 732 best practice guideline for CFD simulation of flows in the urban environment: A summary," *Int. J. Environ. Pollut.* **44**(1–4), 419–427 (2011).
- ⁷⁰J. Smagorinsky, "General circulation experiments with the primitive equations: 1. The basic experiment," *Mon. Weather Rev.* **91**(3), 99–164 (1963).
- ⁷¹Y. Yu, Y. Yang, and Z. Xie, "A new inflow turbulence generator for large eddy simulation evaluation of wind effects on a standard high-rise building," *Build. Environ.* **138**, 300–313 (2018).
- ⁷²Z. Jiang, T. Kobayashi, M. Sandberg, T. Yamanaka, N. Kobayashi, N. Choi, K. Sano, and K. Toyosawa, "Analysis of single-sided ventilation flows of a generic isolated building using particle tracking method in LES simulation," *Build. Environ.* **235**, 110230 (2023).
- ⁷³M. S. Thordal, J. Chr. Bennetsen, and H. H. H. Koss, "Review for practical application of CFD for the determination of wind load on high-rise buildings," *J. Wind Eng. Ind. Aerodyn.* **186**, 155–168 (2019).
- ⁷⁴Y. Nonomura, N. Kobayashi, Y. Tominaga, and A. Mochida, "The cross comparison of CFD results for flow field around building models (part 3)" (in Japanese), *Jpn. Assoc. Wind Eng.* **2003**, 41–41.
- ⁷⁵Y. Tominaga, R. Yoshie, A. Mochida, H. Kataoka, K. Harimoto, and T. Nozu, "Cross comparisons of CFD prediction for wind environment at pedestrian level around buildings," in *The Sixth Asia-Pacific Conference on Wind Engineering (APCWE-VI)*, Seoul, Korea (2005).