Energy Reports 12 (2024) 2621–2630

Contents lists available at ScienceDirect

Energy Reports

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

A deep-learning approach for 3D realization of mean wake flow of marine hydrokinetic turbine arrays

Zexia Zhang [a], Fotis Sotiropoulos [b], Ali Khosronejad [c,*]

[a] Civil Engineering Department, Stony Brook University, Stony Brook, NY 11794, USA
[b] Mechanical & Nuclear Engineering Department, Virginia Commonwealth University, Richmond, VA 23284, USA
[c] Civil Engineering Department, Stony Brook University, Stony Brook, NY 11794, USA

ARTICLE INFO

Keywords:
Marine hydrokinetic turbines
Tidal farms
Wake flow predictions
Large-eddy simulation
Convolutional neural networks

ABSTRACT

We present a novel convolutional neural network (CNN) algorithm to reconstruct turbulence statistics in the wake of marine hydrokinetic (MHK) turbine arrays installed in large meandering rivers. To train the CNN, we utilize large eddy simulation (LES) data depicting the wake flow from a single row of turbines. Once trained, the CNN is deployed to forecast the wake flow of MHK turbine arrays under different hydrodynamic conditions and for varying waterway plan-form geometry. Validation of the CNN predictions are conducted using independently performed LES. Our findings demonstrate the capacity of CNN to accurately predict the wake flow of MHK turbine arrays at significantly reduced computational cost compared to LES. Additionally, the comparison between CNN and unsteady Reynolds-averaged Navier-Stokes (URANS) simulation exhibits a notable advantage of CNN in prediction efficiency and accuracy. This research highlights the potential of CNN to establish reduced-order models for facilitating control co-design and optimization of MHK turbine arrays within natural environments.

1. Introduction

Horizontal-axis marine hydrokinetic (MHK) turbines have been widely used in tidal farms to extract tidal stream energy. Deployment of turbine arrays can increase the efficiency of energy production while minimizing the cost. Due to hydrodynamic interactions between the wakes of individual turbines and hydro-morphodynamics interactions over erodible beds, the layout of turbine arrays not only could affect their performance, but also could impact the ambient aquatic environment and waterway ecosystem (Gotelli et al., 2019). An opportune siting and yaw of turbines can mitigate and even eliminate these effects. Musa et al.(2018) used a quasi-field scaled experiment to demonstrate the impacts of the MHK turbine arrays on the migrating sediment bed, verifying that an appropriate siting of turbines will not jeopardize the river's geomorphic equilibrium and the turbine foundation's structural safety. Musa et al.(2020) used the yawed turbines to deflect the wake away from the riverbank in the experiment, preventing bank erosion with a minimal loss of power production. Modali et al.(2021) studied the near-wake characteristic of the yawed turbine and showed that the available kinetic energy of the downstream turbine could increase by over 50 % by taking the staggered configuration and yawing the upstream turbine. Those researchers highlighted the importance of optimizing the layout of MHK turbine arrays and yaw angles before deployment.

Chawdhary et al. (2017) investigated the wake flow of TriFrame configuration of turbines using large-eddy simulation (LES) with the curvilinear immersed boundary method (CURVIB), revealing the unique wake characteristics of the TriFrame configuration and its advantage in power production over three independent single turbines. Chawdhary et al. (2018) also simulated a 30-turbine MHK turbine array deployed in the East River in New York City using LES with CURVIB, demonstrating the possibility of MHK energy harvesting from large-scale rivers. However, the computational cost of such high-fidelity modeling prohibited their application for array optimization tasks.

Simplified models based on two-dimensional (2D) partial differential equations are more practical for water flow. Zhang et al. (2021) used a 2D tidal stream turbine yaw model based on the shallow water equations and Sequential Least Squares Programming algorithm to optimize the layout and yaw of the tidal turbine array from the aligned arrangement, increasing the energy output by over 80 %. Zhang et al. (2020) optimized the staggered turbine array around Zhoushan Islands using a 2D model based on OpenTidalFarm, increasing the energy extraction by

* Corresponding author.
E-mail address: ali.khosronejad@stonybrook.edu (A. Khosronejad).

https://doi.org/10.1016/j.egyr.2024.08.047
Received 24 March 2024; Received in revised form 14 August 2024; Accepted 17 August 2024
Available online 30 August 2024

6.8 %. However, such 2D models oversimplify the vertical wake structures and replace turbines with the rotor disc model, which neglects the blade effects and provides a low resolution of wake characteristics.

To enable efficient optimization of MHK turbine arrays, González-Gorbeña et al. (2018) developed a surrogate-based optimization (SBO) method based on three-dimensional (3D) numerical experiments. This approach first constructs a surrogate model of design variables using a series of 3D numerical simulation results and then applies mathematical optimization to the surrogate model. However, the number of required numerical simulations is about 10 times the number of design variables; consequently, the computational cost would still be relatively high when more design variables are considered.

Recent advancements in machine-learning techniques have motivated researchers to explore data-driven algorithms for efficient forecasting of complex turbulent flows. For instance, Guo et al. (2016) utilized a convolutional neural network (CNN) autoencoder (CNN-AE) to model the steady flow field around bluff bodies directly, avoiding the costly high-fidelity simulations in steady flow scenarios. Fukami et al. (2019), (2020) employed a hybrid down-sampled skip-connection/-multi-scale CNN algorithm to reconstruct high-resolution flow fields from low-resolution data, which could help to run a less expensive, low-fidelity simulation without sacrificing high-fidelity data. However, the super-resolution approach is still in the very early stages. Zhang et al. (2022a) applied a CNN autoencoder to predict 3D turbulence statistics in a meandering river, which bypassed the long sampling time for time-averaging. The data-driven approaches have also been extended to wind turbine wake predictions. Ti et al. (2020), (2021) developed an artificial neural network (ANN) to forecast velocity deficit and turbulence kinetic energy (TKE) in turbine wakes, and Zhang et al. (2022b) reconstructed time-averaged turbine wake flow and predicted power production curves using CNN. Despite these efforts, there remains a gap in the literature concerning the development and application of data-driven algorithms for predicting MHK turbine farm wakes in large-scale riverine environments.

In this study, we proposed a novel machine learning approach that can more efficiently predict the high-fidelity mean flow field and turbulence statistics of MHK turbine farm wakes in large-scale riverine environments, as shown in Fig. 1. We employ data from large eddy simulation (LES) to develop and validate CNN models for predicting MHK turbine wake flow fields in large meandering rivers. For a given configuration, the LES is executed until initial transient decay and physically meaningful instantaneous flow fields emerge in the simulations. Subsequently, a trained CNN is employed to bypass the long LES computational time required to collect data to obtain converged mean flow and turbulence statistics. The simulation time required to obtain converged statistics can be especially long in brute-force LES of flows characterized by slowly evolving coherent structures, as is the case in large meandering rivers. The instantaneous flow field data are fed into the CNN at the input layer, while the time-averaged data are reconstructed from the output layer. The comparison between the CNN predictions and the LES results revealed that the developed CNN autoencoder algorithms are promising for predicting time-averaged wake flow from hydrokinetic turbines at significantly reduced computational costs compared to brute-force LES.

In addition, we also performed an unsteady Reynolds-averaged Navier-Stokes (URANS) simulation in one of our validation cases to compare the accuracy and efficiency against the trained CNN model. The comparison exhibits a higher efficiency and accuracy of the CNN autoencoder algorithm than URANS based model. Hence, this study demonstrates that the proposed machine learning algorithms enable efficient, high-fidelity predictions of MHK turbine arrays' wake flow in large-scale riverine environments; and thus, they could offer a robust computational framework for tidal farm layout optimization in real-life settings.

This paper is organized as follows. Section 2 presents the numerical models for preparing the training and validation data of the turbulence flow in large-scale rivers with embedded MHK turbines. Section 3 describes the test cases and computational details of the high-fidelity simulations. Section 4 describes the CNN autoencoder algorithm and the training and prediction workflow. Section 5 presents the results and discussion, after which Section 6 presents the conclusion of this study.

2. Numerical model

We used our in-house open-source model, the Virtual Flow Simulator (VFS-Geophysics) code, to generate the simulation data of MHK turbine wake flow fields in large meandering rivers. The VFS-Geophysics has LES and URANS modules to resolve the hydrodynamics of the meandering rivers. The LES module uses the spatially-filtered Navier-Stokes equations and continuity equations as the governing equations to resolve the 3D incompressible turbulent flow, and the URANS module uses the $k-\omega$ model to close the Navier-Stokes equations and continuity equations. To handle the curved geometry of the river, the equations are transformed from Cartesian coordinates $\{x_i\}$ into generalized curvilinear coordinates $\{\xi^i\}$. The non-dimensional form of the equations, in compact tensor notation, reads as follows (Khosronejad and

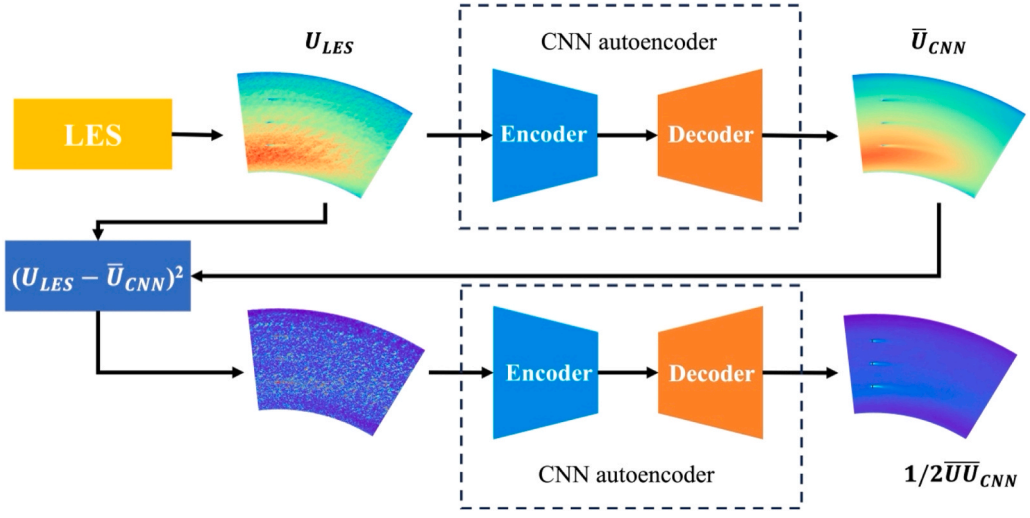

Fig. 1. : Workflow of the proposed CNN autoencoder algorithm. U_{LES} is the instantaneous velocity of the LES computed fully developed turbulent flow. $\bar{U}_{CNN}$ is the CNN predicted time-averaged velocity. $1/2\overline{UU}_{CNN}$ is the CNN predicted turbulence kinetic energy.

Sotiropoulos, 2020):

$$J\frac{\partial U^j}{\partial \xi^j} = 0 \tag{1}$$

$$\frac{\partial U^j}{\partial t} = \xi_l^i\left[-\frac{\partial U^j u_l}{\partial \xi_j} + \frac{1}{\mathrm{Re}}\frac{\partial}{\partial \xi^j}\left(\frac{\xi_l^j \xi_l^k}{J}\frac{\partial u_l}{\partial \xi^k}\right) - \frac{\partial}{\partial \xi^j}\left(\frac{\xi_l^j p}{J}\right) - \frac{\partial \tau_{lj}}{\partial \xi^j} + f_l\right] \tag{2}$$

where $J = \left|\partial(\xi^1,\xi^2,\xi^3)/\partial(x_1,x_2,x_3)\right|$ is the Jacobian of the geometric transformation, $\xi_l^i = \partial\xi^i/\partial x_l$ are the transformation metrics, u_i is the i^{th} Cartesian velocity component, $U^i = (\xi_m^i/J)u_m$ is the contravariant volume flux, $g^{jk} = \xi_l^j\xi_l^k$ are the components of the contravariant metric tensor, p is the pressure, Re is the Reynolds number, and f_l is the body force. In the LES module, τ_{ij} is the sub-grid stress tensor (Kang et al., 2011) modeled using the dynamic Smagorinsky sub-grid scale (SGS) model (Smagorinsky, 1963). In the URANS module, τ_{ij} is the Reynolds stress of the $k-\omega$ model (Wilcox, 1994). The governing equations for the $k-\omega$ model read in generalized curvilinear coordinates as follows (Khosronejad et al., 2011):

$$\frac{1}{J}\frac{\partial(\rho k)}{\partial t} + \frac{\partial}{\partial \xi_j}\left(\rho k U_j\right) = \tau_{ij}\frac{\xi_j^k}{J}\frac{\partial u_i}{\partial \xi_k} - \frac{1}{J}\beta^*\rho k\omega + \frac{\partial}{\partial \xi_j}\left((\mu + \sigma^*\mu_t)\frac{g^{ik}}{J}\frac{\partial k}{\partial \xi_k}\right) \tag{3}$$

$$\frac{1}{J}\frac{\partial(\rho\omega)}{\partial t} + \frac{\partial}{\partial \xi_j}\left(\rho\omega U_j\right) = \alpha\frac{\rho\omega}{k}\tau_{ij}\frac{\xi_j^k}{J}\frac{\partial u_i}{\partial \xi_k} - \frac{1}{J}\beta\rho\omega^2 + \frac{\partial}{\partial \xi_j}\left((\mu + \sigma^*\mu_t)\frac{g^{ik}}{J}\frac{\partial \omega}{\partial \xi_k}\right) \tag{4}$$

$$\mu_t = \frac{\rho k}{\omega} \tag{5}$$

where μ_t is the dynamic eddy viscosity, ρ is the density, and the closure coefficients are given as $\alpha = 5/9$, $\beta = 3/40$, $\beta^* = 9/100$, and $\sigma^* = 1/2$. The Reynolds stress tensor is calculated using the Boussinesq hypothesis as follows (Khosronejad et al., 2011):

$$\tau_{ij} = -2\mu_t S_{ij} + \frac{2}{3}\rho k\delta_{ij} \tag{6}$$

where S_{ij} is the Reynolds averaged strain-rate tensor.

The governing equations are discretized in space on a hybrid staggered/non-staggered grid arrangement using second-order accurate central differencing for the convective terms and second-order accurate, three-point central differencing for the divergence, pressure gradient, and viscous-like terms (Gilmanov and Sotiropoulos, 2005). The time derivatives are discretized using a second-order backward differencing scheme (Kang et al., 2011). The discrete flow equations are integrated in time using an efficient, second-order accurate, fractional step methodology: it is coupled with a Jacobian-free, Newton-Krylov solver for the momentum equations and a generalized minimal residual method (GMRES) solver enhanced with the multigrid method as a preconditioner for the Poisson equation. The hydrodynamic modules of the VFS-Geophysics model have been extensively validated against the laboratory and field scale measurements elsewhere (e.g., see (Kang et al., 2020; Khosronejad et al., 2011, 2020a,b; Zhang et al., 2022a)).

The rotor blades of the MHK turbines are parametrized using the standard actuator line model (ALM) (Yang et al., 2017, 2015). In this approach, the turbine rotor blades are modeled by a straight rotating actuator line. The blades are divided into elements in the radial direction, and the forces on each blade element are calculated based on local inflow velocity, lift, and drag coefficients of 2D airfoils obtained from tabulated files. The interaction between the rotor blades and the flow is represented by forces on the actuator line. In order to take into account the nacelle effect without resolving the surrounding boundary layer, the nacelle is parametrized by the nacelle actuator surface model (ASM) (Yang and Sotiropoulos, 2018). The force acting on the actuator surface is decomposed into normal and tangential components. The tangential force acting on the actuator surface is assumed to be proportional to the local incoming velocity, and the empirical friction coefficient parameterizes the effects of the surface geometry and near-wall turbulence.

3. Test case description and computational details

This section describes the data preparation of the large-scale waterways with embedded MHK turbines for training and validating the proposed CNN algorithm. Two virtual large-scale meandering rivers and three turbine configurations were considered, as shown in Fig. 2. The river geometry is generated using a standard geometric model for the centerlines of meandering rivers (Parker et al., 1983; Abad and Garcia, 2009; Khosronejad et al., 2022). River 1 is 2110 m long, and river 2 is 2740 m long. Both rivers are 100 m wide and 3.3 m deep. MHK turbines are located at the mid-depth of the bend of the waterway. Each MHK turbine has a three-bladed rotor of diameter $D = 1.5m$ and a cylindrical nacelle of 0.3 m diameter. The foil shape of the rotor blade is NACA 4412 hydrofoil. The nondimensional tip speed ratio (TSR) of the turbines is 4. In Fig. 2a, three MHK turbines (marked as solid lines) are aligned and spaced evenly in the spanwise direction at the mid-length of river 1 (1055 m from the inlet). Fig. 2b has two more turbines positioned 16D downstream of the first row in a staggered manner. Fig. 2c shows two rows of turbines at the mid-length of river 2 (the upstream row of turbines is 1370 m from the inlet), spaced evenly in the spanwise direction and 16D in the streamwise direction. The bulk velocity of river 1 and river 2 is 2.04 m s^{-1} and 10.2 m s^{-1}, respectively. The geometry and hydrodynamic details are presented in Table 1.

The flow in the meandering river is solved over the curvilinear structured grid system, which fits the curved geometry of the virtual river. A mesh-independent study is conducted over the training case (Fig. 2a) to determine the grid size. Three grid systems in different resolutions are tested, as shown in Table 2. Grid nodes are distributed uniformly in all three directions. We compared the profile of non-dimensionalized time-averaged velocity magnitude $\overline{U}/U_b$ along the spanwise direction at 3D downstream of the turbine in the mid-depth of the river in Fig. 3. The result demonstrates that the coarser mesh 1 cannot sufficiently resolve the wake of the turbine, but the medium mesh 2 and the finer mesh 3 can provide satisfactory results with the same high accuracy. Therefore, we decided to use the grid resolution of the medium mesh 2, which has a spacing of 0.32, 0.20, and 0.165 m, respectively, in the streamwise, spanwise, and vertical directions, to run the LES of the three cases. Regarding the turbine rotor diameter of 1.5 m, the turbine rotor is resolved by 8 and 9 grid nodes in spanwise and vertical directions, respectively. The simulation time step $\Delta t = 0.05H/U_b$ is elaborated to give a Courant–Friedrichs–Lewy number less than 1. Details of the discretization parameters are shown in Table 3.

The inlet and the outlet of the waterways are carried out using the periodic boundary conditions. The side walls and the flatbed of the waterway apply the no-slip boundary condition. The free surface of the river is described using the rigid-lid assumptions. The rotor blades are parameterized using the ALM, and the nacelle is parameterized using the nacelle ASM.

3.1. The CNN autoencoder for flow reconstruction

The proposed reconstruction algorithm is equivalent to finding a mapping $\mathscr{F}$ from 3D instantaneous snapshots to 3D turbulence statistics field:

$$\left(\overline{u_i},\ \overline{p},\ \overline{u_i'u_j'}\right) = \mathscr{F}(x_i, u_i, p) \tag{7}$$

where x_i denotes the coordinate components, and u_i and p are instantaneous velocity components and pressure from an arbitrary timestep. $\overline{u_i}$ and $\overline{p}$ represent mean velocity components and mean pressure. $\overline{u_i'u_j'}$ are the six components of Reynolds stresses. The mapping $\mathscr{F}$ aims to

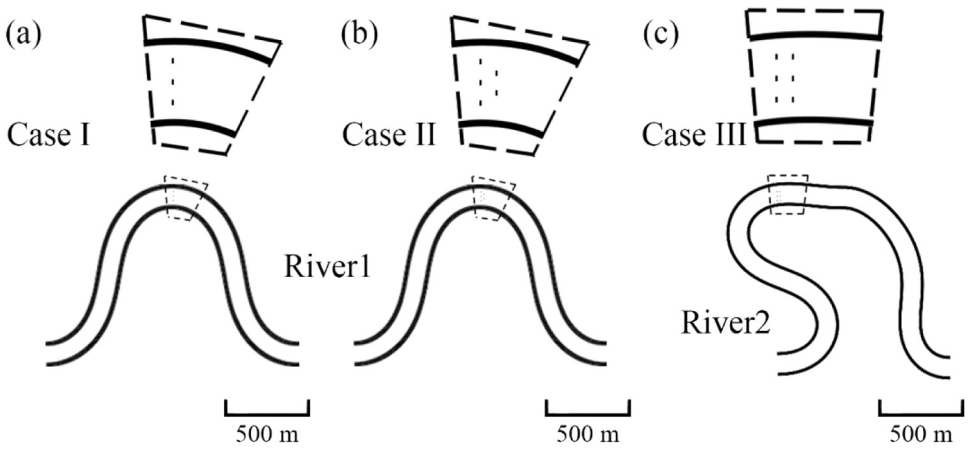

Fig. 2. : Schematic of the virtual meandering river for the training (a) and validation (b) and (c) of the CNN. Solid lines in the virtual river mark the turbines. Flow is from left to right.

Table 1
Geometrical and hydrodynamic characteristics of the virtual testbed rivers. H, B, and L are the mean-flow depth, width, and length of the meandering testbed rivers, respectively. D is the rotor diameter of the MHK turbines. U_b is the bulk velocity of the river. Re is Reynolds numbers.

	River1	River2
H (m)	3.3	3.3
B (m)	100	100
L (m)	2110	2740
D (m)	1.5	1.5
U_b (m s^{-1})	2.04	10.2
Re	6.74×10^{7}	3.37×10^{8}

Table 2
Mesh information of the mesh independence study.

	$N_x \times N_y \times N_z$	Δx (m)	Δy (m)	Δz (m)	Number of nodes
Mesh 1	$5237 \times 397 \times 17$	0.40	0.252	0.206	35,344,513
Mesh 2	$6601 \times 501 \times 21$	0.32	0.2	0.165	69,449,121
Mesh 3	$8317 \times 629 \times 24$	0.254	0.158	0.132	130,784,825

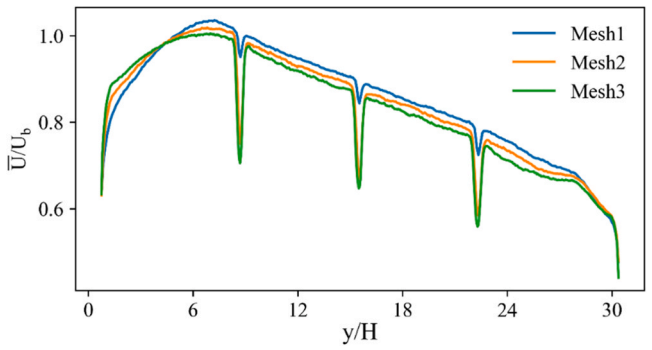

Fig. 3. : Results of the mesh independence study. Profiles are the non-dimensionalized time-averaged velocity magnitude along the spanwise direction at 3D downstream from the turbine in the mid-depth of the river case I. U_b is the bulk velocity. y/H is the distance from the inner bank normalized by the mean flow depth.

minimize the difference between the predicted quantities and the ground truth. In this study, the CNN autoencoder is used as a surrogate model for the mapping $\mathscr{F}$ since it is widely used for its efficiency in processing high-dimensional grid-like data (Santoni et al., 2023; Zhang et al., 2022b, a).

Table 3
Discretization parameters of the meandering rivers. N_x, N_y, and N_z are the number of computational grid nodes in streamwise, spanwise, and vertical directions, respectively. Δx, Δy, and Δz are the spatial resolution in streamwise, spanwise, and vertical directions, respectively. z^+ is the vertical resolution in the wall unit, and Δt is the temporal resolution.

	River1	River2
$N_x \times N_y \times N_z$	$6601 \times 501 \times 21$	$8565 \times 501 \times 21$
Δx (m)	0.32	0.32
Δy (m)	0.20	0.20
Δz (m)	0.165	0.165
z^+	>1000	>1000
Δt	$0.05H/U_b$	$0.05H/U_b$

3.2. CNN autoencoder algorithm

The CNN autoencoder contains two parts: an encoder maps the high-dimensional input field quantities to feature maps in the latent space with reduced dimensions, and a decoder reconstructs the output field quantities from the latent space. The encoder is built by convolutional layers defined by Zhang et al. (2022a) as the following:

$$y_n = \sigma\left(\sum_m k_{m,n} * x_m + b_n\right) \tag{8}$$

where x_m and y_n respectively denote the mth input channel and the nth output channel of the layer, $k_{m,n}$ is the convolutional kernel, $*$ denotes the convolution operator, and b_n is the bias of the nth output channel. Each layer can contain multiple input and output channels that correspond to different input and output feature maps. Specifically, the first layer's input channels correspond to the multiple input quantities. Consequently, $m \times n$ convolutional kernels are applied to extract the nth feature of the mth input channel. Convolutional kernel traverse through the layer's input data with a stride greater than one to reduce the dimension of the feature map. Additionally, a padding is applied to each layer's input to fine tune the output dimension. The convolutional kernel $k_{m,n}$ and the bias b_n are learnable parameters, which the values will be updated during the training process to minimize the objective function. After each convolutional layer, a leaky rectified linear unit (ReLU) function σ is applied to the model, which is given by Maas et al. (2013):

$$\sigma(y) = \begin{cases} y, & y > 0 \\ 0.01y, & y < 0 \end{cases} \tag{9}$$

The decoder is built by transposed convolutional layers which is the

adjoint operation of convolution. The transposed convolutional layers can reconstruct the high-dimensional data from the lower-dimensional feature maps. The input and output dimensions are the reverse of the corresponding convolutional layers.

The training process of the CNN autoencoder is to investigate the optimal set of parameters $k_{m,n}$ and b_n that minimize the objective function. By adopting the naïve data-driven approach, the objective function would be the mean square error (MSE) of the discrepancy between the predicted results and the ground truth, as follows:

$$Loss = MSE(\psi_{CNN} - \psi_{LES}) \tag{10}$$

where ψ_{CNN} and ψ_{LES} are the variables obtained from the CNN and LES, respectively.

The architecture of the CNN autoencoder used in the present study is shown in Fig. 4 and Table 4. The encoder is composed of four 3D convolutional layers (Conv3d). The decoder has three 3D transposed convolutional layers (convT3d) followed by a convolutional layer. A coarse grid search was conducted over the hyperparameter space to determine the number of layers, channels, kernel sizes, and strides by maximizing the generality and performance of the trained model.

3.3. Model training

To produce the training and validation dataset for the CNN, we carried out LES of the three cases shown in Fig. 2. A series of LES were first conducted to produce the fully converged instantaneous flow field. The convergence of instantaneous flow field results was determined by monitoring the time history of the LES-computed total kinetic energy of the flow field. Subsequently, the LES were continued for $t = 800H/U_b$ to time-average the flow field, generating statistically converged turbulence statistics; that is, the mean flow velocity field and Reynolds stresses. The convergence of the time-averaged results was achieved using the time-history analysis method reported in Khosronejad et al. (2020). Then, the instantaneous velocity field snapshots, the time-averaged velocity, and the Reynolds stress fields were used for the training and validation of the CNN. Instantaneous velocity field snapshots were saved at every interval of $\Delta t_{save} = 5H/U_b$, and a total of 80 snapshots of case I were used for training. Because the turbine wake is only a small part of the entire domain, to reduce the training cost and make the trained CNN model focused on the turbine wake, we extracted the results in a subdomain that ranges from $17D$ upstream of the first row of turbines to $90D$ downstream for training. Each subdomain contains $501\times501\times21$ grid nodes along the streamwise, spanwise, and vertical directions, respectively.

The workflow to predict the mean flow field and Reynolds stresses using CNN is shown in Fig. 5. First, the LES generates the instantaneous snapshots of velocity (u, v, w) and pressure field (p). Then, every five consecutive instantaneous LES snapshots were averaged to ensure that all relevant coherent structures at various time scales are incorporated in the training process. We note that a range of coherent flow structures induces such variations: 1) structures present in the incoming turbulent flow, 2) turbine wake meandering, and 3) structures induced by the geometry of the waterway. We performed a series of numerical experiments to determine the minimum number of instantaneous snapshots and their temporal interval separation to minimize the training error of the CNN. The averaged input variables are denoted as u_5, v_5, w_5, p_5. Since the LES results are stored in the curvilinear structured mesh system, the data are 3D matrices which can be directly processed by the convolutional layers. The trainings were conducted using Intel Haswell CPU. The training process utilized 80 training samples from different LES time steps of case I with batch size 4. The Adam optimizer was employed in the proposed algorithm. The learning rate of the training had an initial value of 0.01 with a decay rate of 0.7 in a step size of 400 training epochs. The training ran over 700 epochs until the loss curve plateaued, costing around 64 CPU hours. The validation case I was used as the validation data during the training.

The Reynolds stresses are predicted by a separately trained CNNAE2. In our analysis, compared to predicting the mean velocity fields and Reynolds stresses using a single multi-task CNN, a separately trained CNN for Reynolds stresses requires a smaller model and less training cost

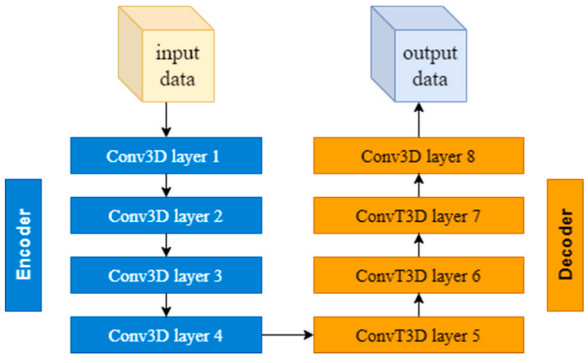

Fig. 4. The flowchart of the CNN autoencoder.

Table 4
Architecture of the CNN autoencoder.

	Layer	Channel	Kernel	Stride	Padding
1	Conv3d	(input, 8)	(3, 3, 3)	(1, 1, 1)	(1, 1, 1)
2	Conv3d	(8, 16)	(3, 3, 3)	(2, 2, 2)	(1, 1, 1)
3	Conv3d	(16, 32)	(3, 3, 3)	(2, 2, 2)	(1, 1, 1)
4	Conv3d	(32, 64)	(3, 3, 3)	(2, 2, 2)	(1, 1, 1)
5	ConvT3d	(64, 32)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
6	ConvT3d	(32, 16)	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
7	ConvT3d	(16, 8)	(4, 4, 4)	(2, 2, 2)	(2, 2, 2)
8	Conv3d	(8, output)	(3, 3, 3)	(1, 1, 1)	(0, 0, 0)

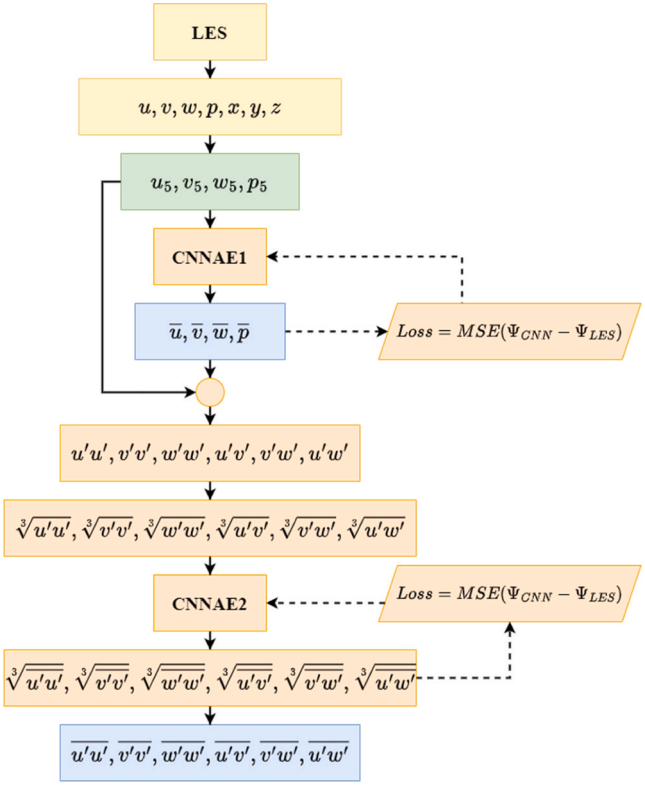

Fig. 5. The workflow of the mean velocity and Reynolds stresses prediction using CNN.

and has a higher predicting accuracy. Different from CNNAE1, the inputs of CNNAE2 are the cross multiplication of velocity fluctuation components. For instance, the input of the streamwise normal stress $u'u'$ is calculated as follows:

$$u'u' = (u_5 - \overline{u}_{CNN})(u_5 - \overline{u}_{CNN}) \tag{11}$$

where the $\overline{u}_{CNN}$ is the predicted time-averaged streamwise velocity component of CNNAE1. Then, the six Reynolds stress components are predicted by CNNAE2. Owing to the nature of the turbulent river flows, the Reynolds stresses in the wake region of the MHK turbines are two orders of magnitude greater than in other areas of the flow. Such significant heterogeneity leads to an unsuccessful initial training of the CNN. To address this issue, we employed a pre-processing approach to render the distribution of second-order statistics more homogeneous. Instead of using $u'u'$nd $\overline{u'u'}$ as the input and the output of CNNAE2, we used their cubic roots: $\sqrt[3]{u'u'}$ and $\sqrt[3]{\overline{u'u'}}$, respectively. Then, the output of CNNAE2 would be post-processed to produce the final results. The training of the CNNAE2 used the same hyperparameters as CNNAE1 and ran over 1200 epochs using around 130 CPU hours.

Next, the trained CNN models were examined by the two validation cases II and III, with different river geometry, Reynolds number, and MHK turbine configurations. The benchmark data for the validation studies were obtained from the LES of the validation cases. The accuracy of the predictions was assessed via the percentage of *relative mean absolute error* (RMAE), defined as such (Zhang et al., 2023):

$$RMAE = \frac{1}{N}\sum_{i=1}^{N}\frac{|\psi_{i(LES)} - \psi_{i(CNN)}|}{|\psi_{i(LES)}|} \times 100\% \tag{12}$$

where $\psi_{i(CNN)}$ and $\psi_{i(LES)}$ are the CNN and LES predicted variables, respectively, at the computational node i, and N is the total number of computational nodes. To further evaluate the performance of the proposed algorithm, we also conducted a URANS simulation on validation case III and compared the URANS results against the prediction results of the proposed algorithm.

4. Results and discussions

4.1. Mean flow predictions

The validation results of the trained CNNAE1 model are shown in Figs. 6 to 9. Figs. 6 and 7 compare the mean velocity magnitude $\overline{U} = \sqrt{\overline{u}^2 + \overline{v}^2 + \overline{w}^2}$ and mean pressure $\overline{p}$ contours of the two validation cases. Four cross sections at 3*D*, 6*D*, 12*D*, and 24*D* downstream of the last row of turbines are marked as I, II, III, and IV, respectively. The top view contours are taken from the hub height level, and the side view contours are taken from the cross sections I and III. As seen in Figs. 6a and 7a, the velocity deficits in the wakes of turbines are captured by the CNN, marking the satisfactory performance of the trained algorithms. To examine the accuracy of the predictions, we illustrate in Figs. 8 and 9 the spanwise profiles of $\overline{U}$ and $\overline{p}$ along the cross sections I to IV at the hub height. Figs. 8a and 9a show that the velocity deficits in the wakes are slightly underestimated. Compared to validation case I, which has the same river geometry and Reynolds number, the overestimation of wake velocity in validation case II is more significant. This is due to the inevitable generalization error of the machine learning method. One possible approach to reduce this discrepancy is to employ more training cases with different turbine configurations. The URANS results of validation case III are also presented in Figs. 7 and 9 for comparison. Comparing to the CNN predictions, the URANS results seriously overestimated the velocity near the inner band (y/H=0). The underestimation of the velocity deficits in the wakes is also more significant.

Moreover, Table 5 presents the prediction errors of the CNN and URANS compared to the LES results for validation cases II and III. Generally, the prediction errors of CNN are lower than those of URANS, which marks the better performance of the CNN algorithm.

4.2. Prediction of turbulence statistics

A separate CNNAE2 was trained to predict the second-order statistics of the turbulent wake flows using the instantaneous velocity fluctuations calculated from the LES instantaneous velocity snapshots, and the CNN

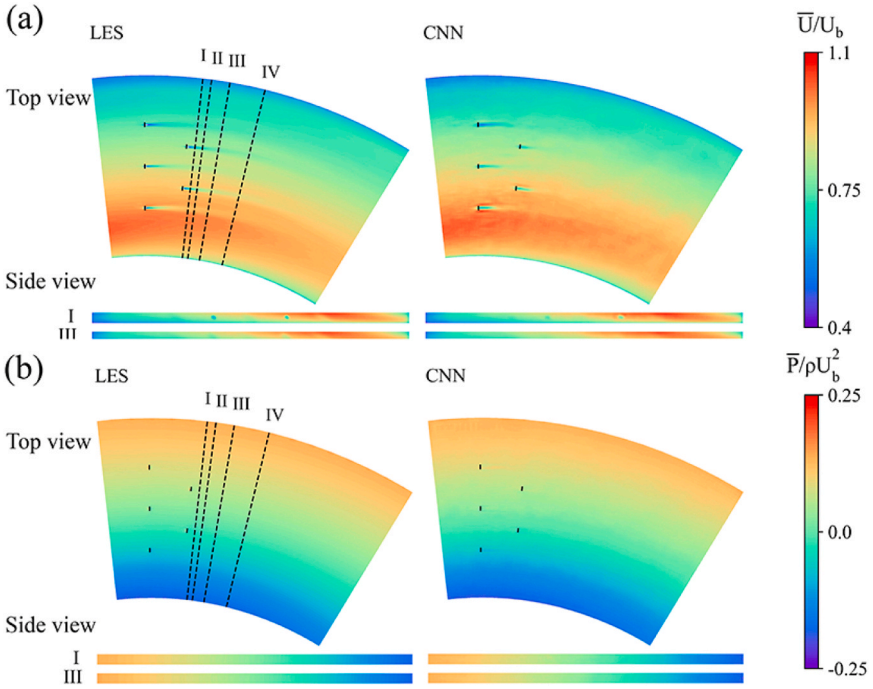

Fig. 6. Contours of the normalized mean velocity and mean pressure of validation case II. The present subdomain is extracted from 17*D* upstream of the first row of turbines to 90*D* downstream. Top views are taken from the hub height. Side views are taken from cross-sections at 6D and 24D downstream of the last row of turbines, marked by the dashed lines I and III. The bulk velocity U_b is 2.04 m s^{-1}. The density of water ρ is 1000 kg m^{-3}. Flow is from left to right.

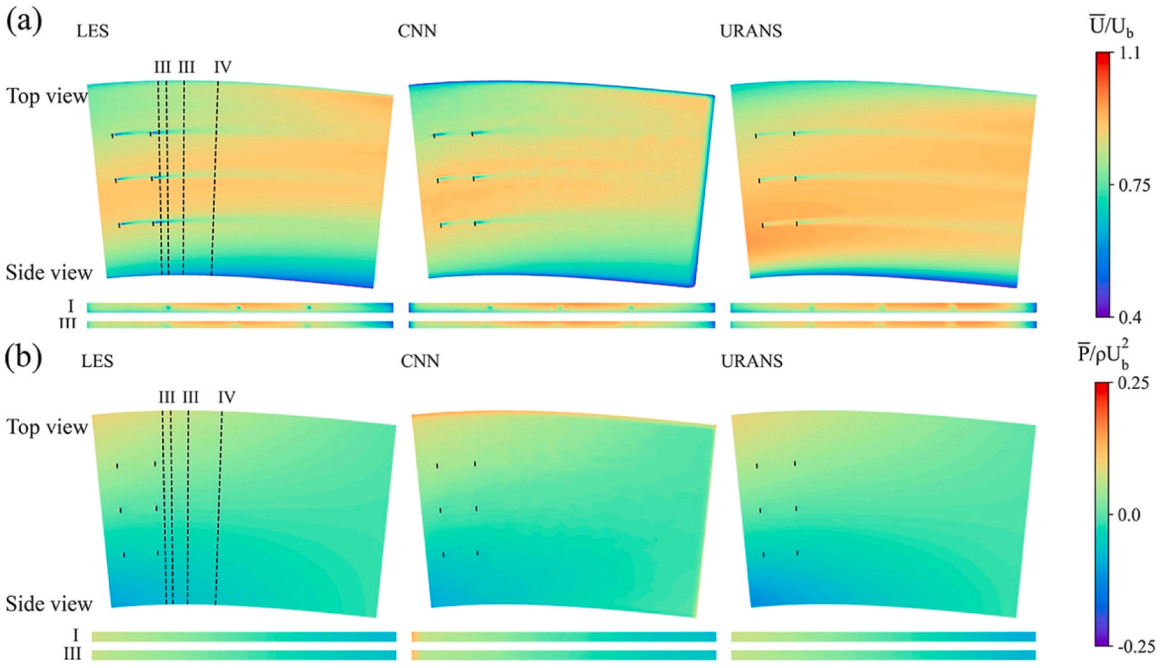

Fig. 7. Contours of the normalized mean velocity and mean pressure of the validation case III. The present subdomain is extracted from 5*D* upstream of the first row of turbines to 90*D* downstream. Top views are taken from the hub height. Side views are taken from cross-sections at 6D and 24D downstream of the last row of turbines, marked by the dashed lines I and III. The bulk velocity U_b is 2.04 m/s. The density of water ρ is 1000 kg/m^3. Flow is from left to right.

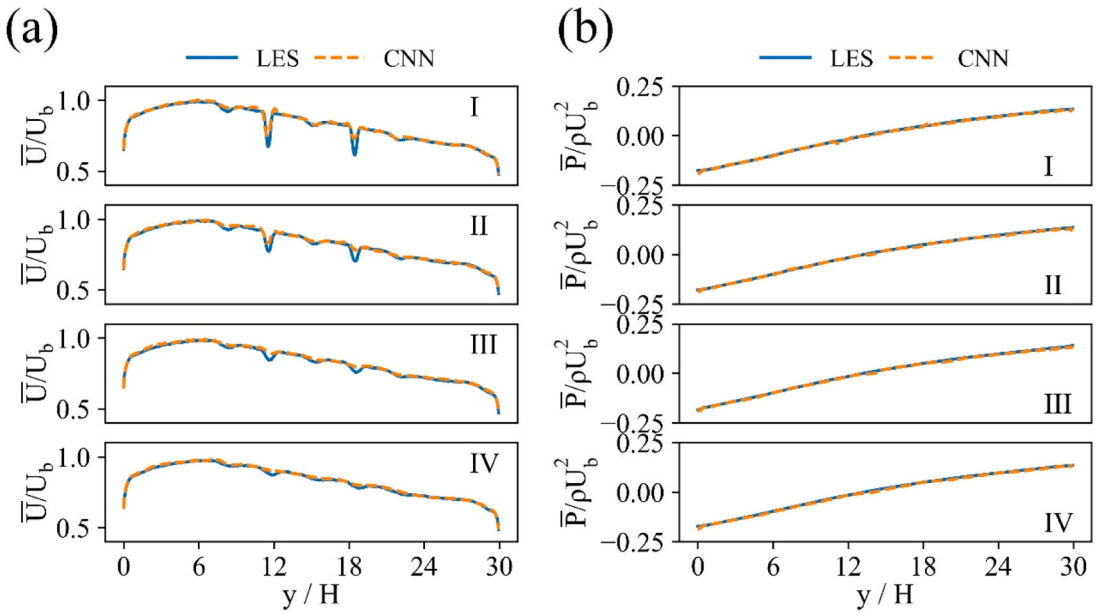

Fig. 8. Profiles of the normalized mean velocity and mean pressure of validation case II. Profiles are along the dashed lines marked by I, II, III, and IV in the contours of the corresponding case in Fig. 6, which are at 3*D*, 6*D*, 12*D*, and 24*D* downstream the last row of turbines, respectively. The bulk velocity U_b is 2.04 m/s. The density of water ρ is 1000 kg/m^3. y/H is the distance from the inner bank normalized by the mean flow depth H of 3.3 m.

predicted mean velocity. For simplicity, instead of plotting all six Reynolds stresses, the contours of the TKE obtained from the predicted Reynolds stresses (TKE = $1/2(\overline{u'u'}+\overline{v'v'}+\overline{w'w'})$) by the CNN were compared against the LES results in Figs. 10 and 11. As seen in Fig. 10, the CNN has successfully reconstructed the TKE in the wakes of MHK turbines in both validation cases. The profiles across the width of the waterway in Fig. 11 show that the high TKE peaks were accurately captured in both validation cases. In comparison, the URANS result of validation case III significantly underestimated the TKE in the wakes. The CNN has also accurately reconstructed the TKE distribution in the background flow although the variation is infinitesimal compared to the TKE in the wakes. This impressive performance is attributed to the cubic root preprocessing. In comparison, the URANS neglected the background TKE distribution. The advantage of the CNN over URANS is also demonstrated in Table 6. The relative error of the CNN prediction is about half of the URANS results of validation case II. We also note that the relative error of case III is higher than case II because of the overestimation of the TKE near the inner bank in the background flow, as seen in the range of y/H=0–6 in Fig. 11. This overestimation was caused by the generalization error in different rivers and could be reduced by

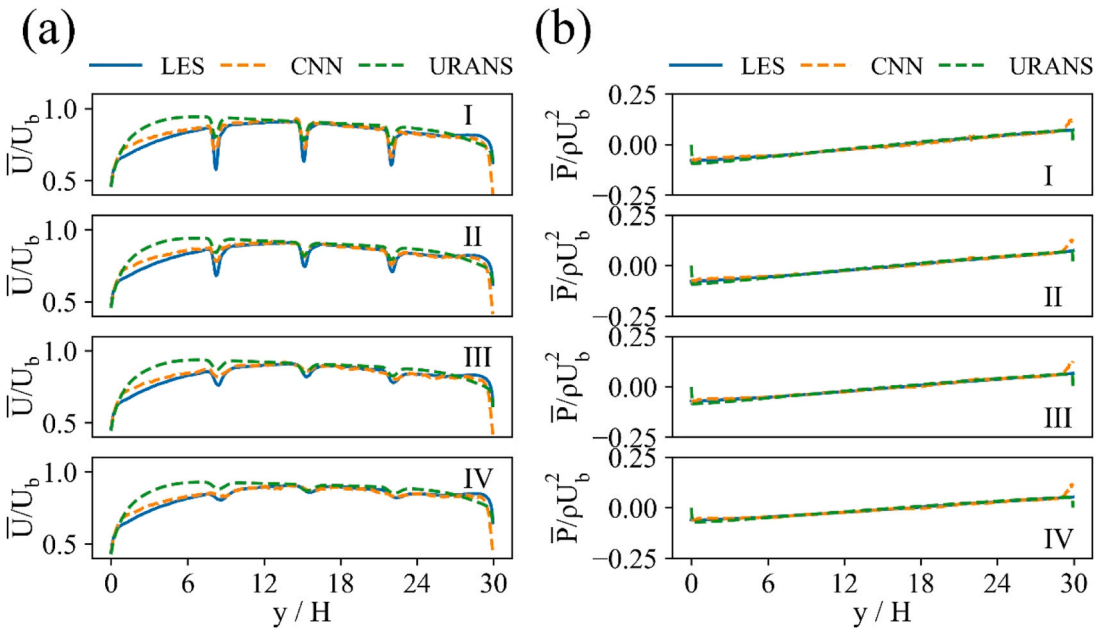

Fig. 9. Profiles of the normalized mean velocity and mean pressure of validation case III. Profiles are along the dashed lines marked by I, II, III, and IV in the contours of the corresponding case in Fig. 7, which are at 3*D*, 6*D*, 12*D*, and 24*D* downstream the last row of turbines, respectively. The bulk velocity U_b is 2.04 m s^{-1}. The density of water ρ is 1000 kg m^{-3}. y/H is the distance from the inner bank normalized by the mean flow depth H of 3.3 m.

Table 5
RMAE for the first-order turbulence statistics predicted by CNN and URANS in the two validation cases II and III. $\overline{U}$ is the time-averaged velocity magnitude. U_b is the bulk velocity of the river. $\overline{p}$ is the mean pressure. ρ is the density of water.

		CNN	URANS
Validation II	$\overline{U}$ / U_b	0.91 %	-
	$\overline{p}$ / ρU_b^2	4.93 %	-
Validation III	$\overline{U}$ / U_b	3.18 %	5.85 %
	$\overline{p}$ / ρU_b^2	18.32 %	18.10 %

including more river geometries in training samples. However, this overestimation did not affect the prediction of the wakes.

In conclusion, the trained CNN can successfully predict the 3D wake flow fields of MHK turbine arrays in large-scale meandering rivers. The mean velocity, mean pressure, and TKE profiles obtained from the CNN predictions agree well with the LES results. And the statistical error indices demonstrated the accuracy of the CNN predictions.

Now, we focus our attention on the computational cost of these wake flow predictions. On average, brute-force LES requires over 3.5×10^4 CPU hrs to generate the converged time-averaged wake flow field of each case, and URANS requires around 1.08×10^4 CPU hrs. The trained CNN algorithms, however, requires about 60 CPU sec to reconstruct the wake flow fields. Considering the training cost of about 120 CPU hrs and the cost of LES to produce instantaneous flow fields (i.e., 0.95×10^3 CPU hrs) for the training inputs of the CNN, the total computational cost of the proposed CNN is about 0.97×10^3 CPU hrs, which is a small fraction of that of brute-force LES or URANS. Therefore, given the accuracy of the proposed CNN algorithms and their computational costs, the proposed approach could enable reliable and affordable predictions of the wake flow field of large-scale tidal farms, and provide efficient prediction tools to optimize the layouts of the MHK turbine arrays in real-life sites.

5. Conclusion

We examined the capabilities of the CNN autoencoder algorithms to predict the turbulence statistics of the 3D wake flow field of MHK turbine arrays in a large-scale meandering river. Three turbine configurations in two different virtual meandering rivers were considered as the training and validation cases for the CNN. The training case was a single row of three turbines located at the apex of the meander. In contrast, the two validation cases were two rows of turbines with aligned and staggered wakes in two different rivers and under different Reynolds numbers. The LES with ALM of the three cases were conducted to produce the training and validation dataset of the CNN. A CNN is first trained to predict the first-order turbulence statistics of the tidal farms' wake flow field. To do so, we considered the LES-computed instantaneous and time-averaged flow fields as the CNN algorithm's input and output arrays, respectively.

Furthermore, a separate eight-layer CNN was trained to predict the second-order turbulence statistics using the inputs derived from the predictions of CNN. In the training process, the inputs came from cross multiplication of velocity fluctuations derived from the instantaneous LES results and the CNN-predicted time-averaged results. At the same time, the LES-computed normal Reynolds stresses were set as the output. The trained CNN was then employed to predict the second-order turbulence statistics of the validation cases. The CNN's predictions were compared against the time-averaged flow field results of separately done LES and URANS. The trained CNN can successfully predict the 3D wake flow fields of MHK turbine arrays in large-scale meandering rivers. The overall computational costs associated with the CNN algorithms is roughly 2.8 % of the LES. Compared to URANS, the proposed CNN algorithms can generate the time-averaged flow field with higher accuracy using fewer computational resources, demonstrating its advantage over the URANS approach. Therefore, the proposed CNN algorithms provide efficient prediction tools to optimize the layouts of the MHK turbine arrays in real-life sites.

CRediT authorship contribution statement

Fotis Sotiropoulos: Writing – review & editing, Software, Resources, Methodology, Investigation, Conceptualization. **Ali Khosronejad:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Zexia Zhang:**

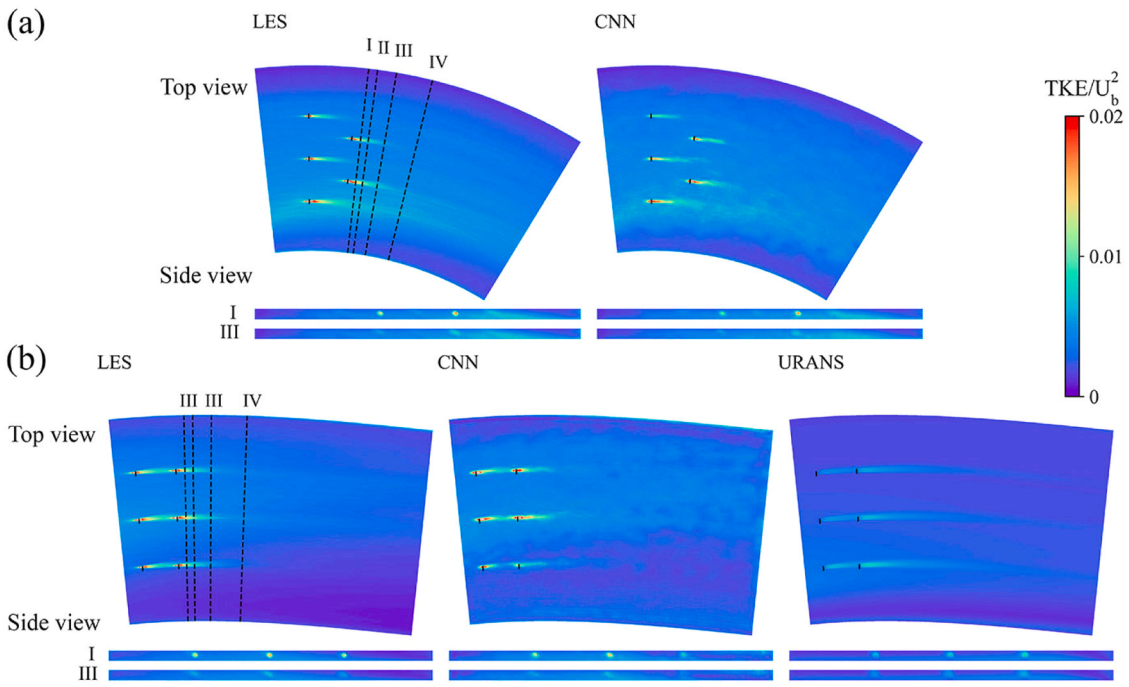

Fig. 10. Contours of the normalized TKE of (a) validation case II and (b) validation case III. The presented subdomain is extracted from (a) 17D upstream of the first row of turbines to 90D downstream, and from (b) 5D upstream of the first row of turbines to 90D downstream. Top views are taken from the hub height. Side views are taken from cross-sections at 6D and 24D downstream the last row of turbines, marked by the dashed lines I and III. The bulk velocity U_b is 2.04 m s^{-1}. Flow is from left to right.

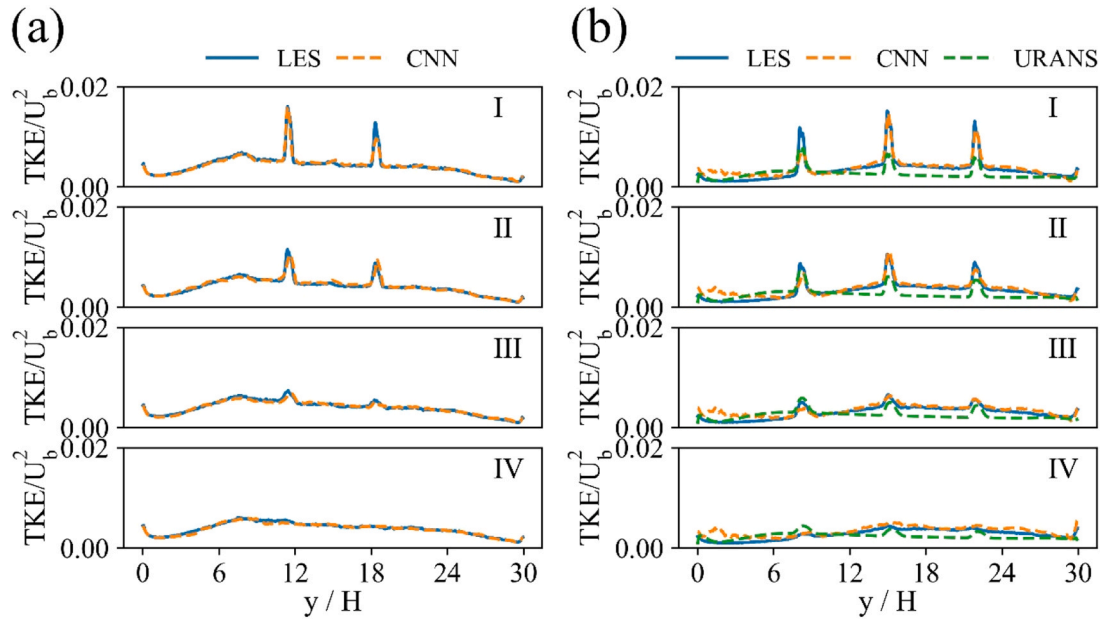

Fig. 11. Profiles of the normalized TKE of (a) validation case I and (b) validation case II. Profiles are along the dashed lines marked by I, II, III, and IV in the contours of the corresponding case in Fig. 10, which are at 3D, 6D, 12D, and 24D downstream of the last row of turbines, respectively. The bulk velocity U_b is 2.04 m s^{-1} y/H is the distance from the inner bank normalized by the mean flow depth H of 3.3 m.

Table 6
RMAE for the 2nd order turbulence statistics predicted by CNN and RANS in the two validation cases II and III. TKE is the turbulence kinetic energy. U_b is the bulk velocity of the river.

		CNN	URANS
Validation II	$TKE/\ U_b^2$	5.06 %	-
Validation III	$TKE/\ U_b^2$	17.49 %	34.00 %

Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by a grant from the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Water Power Technologies Office (WPTO) Award Number DE-EE0009450. Partial support was also provided by NSF (grant number 2233986). The computational resources were provided by the Civil Engineering Department, Stony Brook Research Computing and Cyber-infrastructure, and the Institute for Advanced Computational Science at Stony Brook University. We thank Jonathan Craig for his contribution on improving the language and readability of this manuscript.

References

Abad, J.D., Garcia, M.H., 2009. Experiments in a high-amplitude Kinoshita meandering channel: 1. Implications of bend orientation on mean and turbulent flow structure. Water Resour. Res. 45 https://doi.org/10.1029/2008WR007016.

Chawdhary, S., Angelidis, D., Colby, J., Corren, D., Shen, L., Sotiropoulos, F., 2018. Multiresolution large-eddy simulation of an array of hydrokinetic turbines in a field-scale river: the Roosevelt island tidal energy project in New York City. Water Resour. Res. 54, 10,188-10,204. https://doi.org/10.1029/2018WR023345.

Chawdhary, S., Hill, C., Yang, X., Guala, M., Corren, D., Colby, J., Sotiropoulos, F., 2017. Wake characteristics of a TriFrame of axial-flow hydrokinetic turbines. Renew. Energy 109, 332–345. https://doi.org/10.1016/j.renene.2017.03.029.

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

Fukami, K., Fukagata, K., Taira, K., 2020. Assessment of supervised machine learning methods for fluid flows. Theor. Comput. Fluid Dyn. 34, 497–519. https://doi.org/10.1007/s00162-020-00518-y.

Gilmanov, A., Sotiropoulos, F., 2005. A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies. J. Comput. Phys. 207, 457–492. https://doi.org/10.1016/j.jcp.2005.01.020.

González-Gorbeña, E., Qassim, R.Y., Rosman, P.C.C., 2018. Multi-dimensional optimisation of tidal energy converters array layouts considering geometric, economic and environmental constraints. Renew. Energy 116, 647–658. https://doi.org/10.1016/j.renene.2017.10.009.

Gotelli, C., Musa, M., Guala, M., Escauriaza, C., 2019. Experimental and numerical investigation of wake interactions of marine hydrokinetic turbines. Energy 12. https://doi.org/10.3390/en12163188.

Guo, X., Li, W., Iorio, F., 2016. Convolutional neural networks for steady flow approximation. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 13-17-Augu, 481–490. https://doi.org/10.1145/2939672.2939738.

Kang, S., Khosronejad, A., Hill, C., Sotiropoulos, F., 2020. Mean flow and turbulence characteristics around multiple-arm instream structures and comparison with single-arm structures. J. Hydraul. Eng. 146, 04020030 https://doi.org/10.1061/(asce)hy.1943-7900.0001738.

Kang, S., Lightbody, A., Hill, C., Sotiropoulos, F., 2011. High-resolution numerical simulation of turbulence in natural waterways. Adv. Water Resour. 34, 98–113. https://doi.org/10.1016/j.advwatres.2010.09.018.

Khosronejad, A., Flora, K., Kang, S., 2020. Effect of inlet turbulent boundary conditions on scour predictions of coupled LES and morphodynamics in a field-scale river: bankfull flow conditions, 4020020 J. Hydraul. Eng. 146. https://doi.org/10.1061/(asce)hy.1943-7900.0001719.

Khosronejad, A., Kang, S., Borazjani, I., Sotiropoulos, F., 2011. Curvilinear immersed boundary method for simulating coupled flow and bed morphodynamic interactions due to sediment transport phenomena. Adv. Water Resour. 34, 829–843. https://doi.org/10.1016/j.advwatres.2011.02.017.

Khosronejad, Ali, Kang, S., Farhadzadeh, A., Sotiropoulos, F., 2020. On the genesis and evolution of barchan dunes: hydrodynamics. Phys. Fluids 32. https://doi.org/10.1063/5.0015515.

Khosronejad, A., Limaye, A.B., Zhang, Z., Kang, S., Yang, X., Sotiropoulos, F., 2022. On the morphodynamics of a wide class of large-scale meandering rivers: Insights gained by coupling LES with sediment-dynamics.

Khosronejad, A., Sotiropoulos, F., 2020. A short note on the simulation of turbulent stratified flow and mobile bed interaction using the continuum coupled flow and morphodynamics model. Environ. Fluid Mech. https://doi.org/10.1007/s10652-020-09752-8.

Maas, A.L., Hannun, A.Y., Ng, A.Y., 2013. Rectifier nonlinearities improve neural network acoustic models.

Modali, P.K., Vinod, A., Banerjee, A., 2021. Towards a better understanding of yawed turbine wake for efficient wake steering in tidal arrays. Renew. Energy 177, 482–494. https://doi.org/10.1016/j.renene.2021.05.152.

Musa, M., Hill, C., Sotiropoulos, F., Guala, M., 2018. Performance and resilience of hydrokinetic turbine arrays under large migrating fluvial bedforms. Nat. Energy 3, 839–846. https://doi.org/10.1038/s41560-018-0218-9.

Musa, M., Ravanelli, G., Bertoldi, W., Guala, M., 2020. Hydrokinetic turbines in yawed conditions: toward synergistic fluvial installations. J. Hydraul. Eng. 146 https://doi.org/10.1061/(asce)hy.1943-7900.0001707.

Parker, G., Diplas, P., Akiyama, J., 1983. Meander bends of high amplitude. J. Hydraul. Eng. 109, 1323–1337. https://doi.org/10.1061/(ASCE)0733-9429(1983)109:10(1323).

Santoni, C., Zhang, Z., Sotiropoulos, F., Khosronejad, A., 2023. A data-driven machine learning approach for yaw control applications of wind farms. Theor. Appl. Mech. Lett. 13 https://doi.org/10.1016/j.taml.2023.100471.

Smagorinsky, J., 1963. General circulation experiments with the primitive equations. Mon. Weather Rev. 91, 99–164 https://doi.org/10.1175/1520-0493(1963)091<0099:gcewtp>2.3.co;2.

Ti, Z., Deng, X.W., Yang, H., 2020. Wake modeling of wind turbines using machine learning. Appl. Energy 257. https://doi.org/10.1016/j.apenergy.2019.114025.

Ti, Z., Deng, X.W., Zhang, M., 2021. Artificial neural networks based wake model for power prediction of wind farm. Renew. Energy 172, 618–631. https://doi.org/10.1016/j.renene.2021.03.030.

Wilcox, D.C., 1994. Simulation of transition with a two-equation turbulence model. AIAA J. 32, 247–255. https://doi.org/10.2514/3.59994.

Yang, X., Khosronejad, A., Sotiropoulos, F., 2017. Large-eddy simulation of a hydrokinetic turbine mounted on an erodible bed. Renew. Energy 113, 1419–1433. https://doi.org/10.1016/j.renene.2017.07.007.

Yang, X., Sotiropoulos, F., Conzemius, R.J., Wachtler, J.N., Strong, M.B., 2015. Large-eddy simulation of turbulent flow past wind turbines/farms: the virtual wind simulator (VWiS). Wind Energy 18, 2025–2045. https://doi.org/10.1002/we.1802.

Yang, X., Sotiropoulos, F., 2018. A new class of actuator surface models for wind turbines. Wind Energy 21, 285–302. https://doi.org/10.1002/we.2162.

Zhang, Z., Flora, K., Kang, S., Limaye, A.B., Khosronejad, A., 2022a. Data-driven prediction of turbulent flow statistics past bridge piers in large-scale rivers using convolutional neural networks. Water Resour. Res. 58 https://doi.org/10.1029/2021WR030163.

Zhang, Z., Hao, X., Santoni, C., Shen, L., Sotiropoulos, F., Khosronejad, A., 2023. Toward prediction of turbulent atmospheric flows over propagating oceanic waves via machine-learning augmented large-eddy simulation. Ocean Eng. 280 https://doi.org/10.1016/j.oceaneng.2023.114759.

Zhang, C., Kramer, S., Angeloudis, A., Zhang, J., Lin, X., Piggott, M.D., 2021. Improving tidal turbine array performance through the optimisation of layout and yaw angles. : Proc. 14th Eur. Wave Tidal Energy Conf. 1–7.

Zhang, Z., Santoni, C., Herges, T., Sotiropoulos, F., Khosronejad, A., 2022b. Time-averaged wind turbine wake flow field prediction using autoencoder convolutional neural networks. Energy 15. https://doi.org/10.3390/en15010041.

Zhang, C., Zhang, J., Tong, L., Guo, Y., Zhang, P., 2020. Investigation of array layout of tidal stream turbines on energy extraction efficiency. Ocean Eng. 196 https://doi.org/10.1016/j.oceaneng.2019.106775.