

# Comparison between conventional and deep learning-based surrogate models in predicting convective heat transfer performance of U-bend channels

Qi Wang<sup>a,b</sup>, Weiwei Zhou<sup>a,b</sup>, Li Yang<sup>a,b,\*</sup>, Kang Huang<sup>b,c</sup>

<sup>a</sup> School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

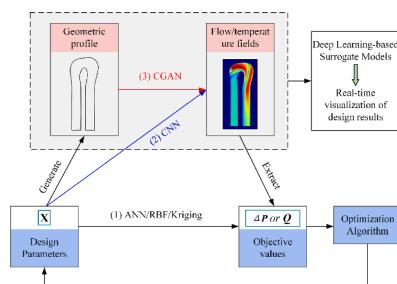
<sup>b</sup> State Key Laboratory of Aerodynamics, Mianyang, Sichuan, 621000, China

<sup>c</sup> China Aerodynamics Research and Development Center, Mianyang, Sichuan, 621000, China

## HIGHLIGHTS

- A large-scale dataset containing 3160 samples is generated for model analysis.
- The deep learning-based models can accurately reconstruct physical fields.
- The conventional models have higher single-point prediction accuracy.
- The deep learning-based models have higher response accuracy.
- The sensitivity of different models to sample size is evaluated.

## GRAPHICAL ABSTRACT



## ARTICLE INFO

**Keywords:**  
Surrogate modeling  
Deep learning  
Convective heat transfer  
U-bend

## ABSTRACT

Deep neural networks are efficient methods to achieve real-time visualization of physics fields. The main concerns that prevented deep learning from being implemented in the field of energy conversion were the risks of overfitting and the lack of data. Therefore, it is necessary to evaluate different kinds of surrogate modeling methods and provide guidelines for designers to choose models. In this study, three conventional models (Artificial Neural Network, Radial Bias Function, and Kriging), and two deep learning-based models (Convolutional Neural Network and Conditional Generative Adversarial Neural Network) were established to predict the flow and heat transfer performance of a U-bend with variable geometries. The models were detailedly compared in terms of the single-point prediction accuracy, response accuracy, sensitivity to sample size, and other characteristics of interest. Results showed that the conventional models had slightly higher single point accuracy and the relative error of pressure loss and heat transfer were within  $\pm 6.6\%$  and  $\pm 5.7\%$  respectively, while those of the deep learning-based models were within  $\pm 8.0\%$  and  $\pm 6.3\%$  respectively. Nevertheless, the deep learning-based models had higher response accuracy and could reconstruct the distributions of surface pressure and wall heat flux with the pixel-wise absolute error within  $\pm 2.0$  Pa and  $\pm 45$  W/m<sup>2</sup> respectively. The results indicated that deep learning was a promising surrogate modeling approach due to its acceptable prediction error and ability to reconstruct physical fields. This effort was expected to serve as a guide for establishing more reliable data-driven surrogate models for energy conversion and heat transfer problems.

\* Corresponding author.

E-mail address: [liy59@sjtu.edu.cn](mailto:liy59@sjtu.edu.cn) (L. Yang).

## Nomenclature

$D_h$	Hydraulic diameter, [m]
$\rho$	Density, [ $\text{kg}/\text{m}^3$ ]
$U$	Velocity, [ $\text{m}/\text{s}$ ]
$U_0$	Bulk flow velocity, [ $\text{m}/\text{s}$ ]
$P$	Pressure, [Pa]
$P^*$	Normalized pressure, [-]
$\Delta P$	Pressure drops, [Pa]
$\Delta P^*$	Normalized pressure drops, [-]
$q$	Wall heat flux, [ $\text{W}/\text{m}^2$ ]
$q^*$	Normalized wall heat flux, [-]
$Q$	Heat transfer, [W]
$RMSE$	Root Mean Squared Error, [-]
$MAE$	Mean Absolute Error, [-]
$r$	Pearson correlation coefficient, [-]

## Introduction

Heat transfer analysis is an important step for designing thermal energy conversion devices. Most thermal designs require iterative structural modifications until specific performance is achieved. The designers generally follow a manual trial-and-error approach, which is extremely time-consuming when there are a large number of parameters to adjust. To make the design process more efficient, optimization methods such as the Genetic Algorithm were applied to automatically adjust design variables.

Surrogate models are the core of cost-effective optimization processes, which mappings between the design variables and objective values. A reliable surrogate model should have the capability of evaluating the objectives of any given design accurately in a short time. Within such a definition, experimental and Computational Fluid Dynamics (CFD) techniques are not suitable methods for optimization due to the high computational cost. Therefore, data-driven methods are widely used for surrogate modeling in many thermal-related engineering applications, such as proton exchange membrane fuel cells [1, 2], air-cooled condensers [3], combustion systems [4], and building energy systems [5].

### Parameter-based surrogate modeling for heat transfer

Conventional surrogate modeling methods established mappings between a collection of zero-dimensional design variables and zero-dimensional objective parameters. Widely used parameter-based models in the past few decades included: Artificial Neural Networks (ANN), Radial Basis Functions (RBF), and Kriging model.

Many studies have been conducted on structural optimizations using these surrogate modeling methods. Li et al. [6] proposed an ANN to predict the area-averaged adiabatic film cooling effectiveness on the external surface of a turbine vane. The established ANN showed good prediction performance. Their work aimed to achieve an ANN-aided design of the film cooling structures. Maral et al. [7] conducted an aerothermal optimization of tip carving for an axial turbine blade using the Genetic algorithm (GA). They tested three surrogate models and the ANN yielded the best prediction performance. Moon et al. [8] presented a multi-objective optimization for a fan-shaped pin-fin in a cooling channel using the GA and the RBF surrogate modeling method to simultaneously maximize the heat transfer and minimize the friction loss. Different pin-fin shapes with performance improvements were obtained under different objective functions. However, only two geometric parameters of the fan-shaped pin-fin were selected as the design variables. Similarly, Wang et al. [9] used the RBF surrogate modeling method and the GA to optimize the shape of a fan-shaped film cooling

hole. The objective was to maximize the area-averaged adiabatic film cooling effectiveness. Their work presented that RBF coupled with GA was an effective optimization scheme. However, only the incline angle, lateral expansion angle and hole length were selected as the design variables.

Namgoong et al. [10] used the Design of Experiment (DOE) and surrogate methods to optimize a U-bend shaped turbine cooling passage with minimum pressure loss. They presented the effectiveness of the surrogate model-based optimization framework. The results showed that a proper surrogate model was important in the optimization process. It was found that the Kriging model performed best in their study. Verstraete et al. [11] utilized a Kriging-assisted differential evolution algorithm for similar purposes. The U-bend was parameterized using piece-wise Bezier curves thus ensuring the wide variety of shapes. The total pressure loss had been reduced by 37.6% compared to the baseline geometry. The optimization results were further experimentally validated [12]. Moreover, Verstraete et al. [13] conducted a multi-objective optimization with Kriging-assisted to simultaneously minimize the pressure loss of the U-bend and maximize the heat transfer. Several trade-off results with significant performance improvements had been obtained. In addition to U-bend, the Kriging model-based optimization technique was also used to optimize the shape of dimpled surfaces [14–16], ribbed surfaces [17], film cooling holes [18, 19], and other structures [20] to maximize the heat transfer.

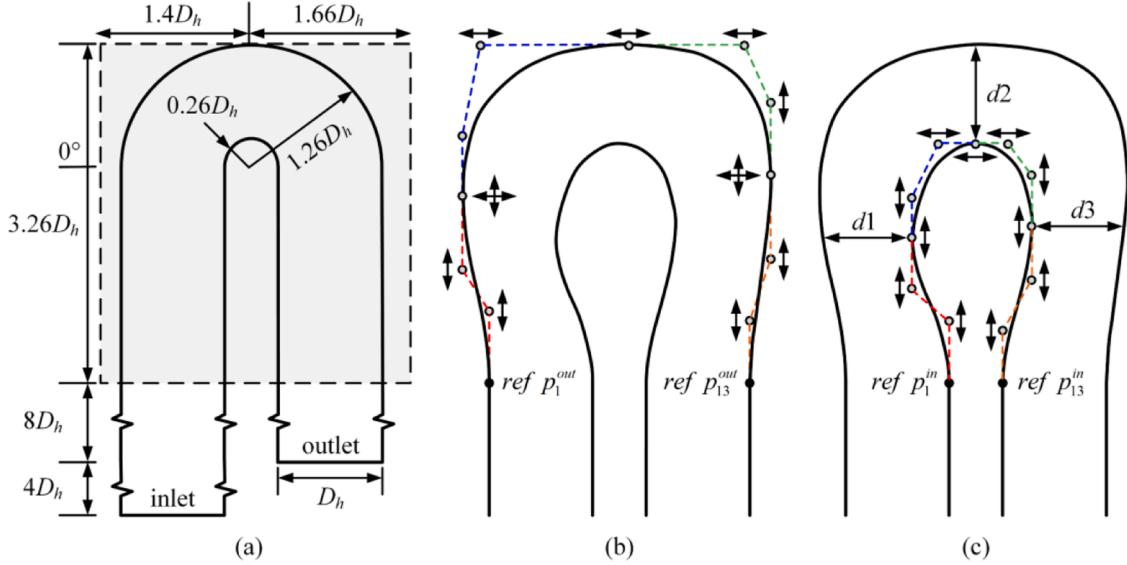
Contemporarily, parameter-based models and parametric design are general techniques for many heat transfer applications. However, from the perspective of complex structure design or topological design, the conventional models still had the following deficiencies: 1) the conventional methods could not provide the information of the physical fields; 2) the parameter-based models are black boxes for designers to interact with. Therefore, the heat transfer domain has been seeking new methods to obtain high-dimensional physical information in real-time.

### Image-based deep learning modeling for heat transfer

Deep learning is a class of advanced data modeling approaches developed in recent years. Due to its strong capability in modeling nonlinearities and modeling high-dimensional information, deep learning could serve as a good Human-Computer Interaction tool for complex heat transfer designs. Recent efforts that crossed heat transfer and deep learning were devoted to temperature or flow field visualization.

As a preliminary attempt, Yang et al. [21, 22] pioneered the use of the convolution modeling method to predict the adiabatic cooling effectiveness of transpiration cooling. They presented the effectiveness of convolution methods in modeling high-dimensional physical information. As compared with conventional correlations, the proposed model provided extensive details of the cooling effectiveness distribution while consuming computational time as short as correlations. However, the model was trained and tested with no more than ten samples and the response of the model was not evaluated. To further improve the modeling ability, Yang et al. [23] established a more advanced model based on Conditional Generative Adversarial Networks (CGAN) [24] to predict the surface temperature distributions of a series of effusively cooled plates. The model took the image of the cooling hole distribution and the image of the surface temperature distribution as input and output respectively and was capable of quickly providing the surface temperature under variable cooling hole distributions. The results presented the great potential of CGAN for 2-D physical field prediction under given geometry and boundary conditions. However, only 20 samples were generated for model training and testing due to high computational cost. In addition, there was still a lack of evaluations of model responses.

More recently, Ma et al. [25] utilized Convolutional Neural Networks (CNN) to predict the 2-D flow field and temperature field of film cooling in a rocket combustor. The flow conditions of the mainstream and



**Fig. 1.** A schematic diagram of the parameterized U-bend. (a) baseline U-bend; (b) parameterization of the outer curve and (c) parameterization of the inner curve.

coolant were encoded in a three-channel image for the input of the model. The model output the flow field and concentration information. Their work presented the capability of CNN to model complicated flow and heat transfer problems. The influence of sample size on the model performance was investigated. However, the sample size was limited to 500. Similarly, Laubscher et al. [26] proposed a model that integrated convolutional variational autoencoders and multi-layer neural networks to predict the temperature, velocity, and species distributions in a methane-fired turbulent jet diffusion flame. They studied the impact of different model architectures and hyperparameters on the model performance but did not consider the impact of sample size. Liu et al. [27] developed a CNN-based model to reconstruct the flow and temperature field in a grooved microchannel. They considered two kinds of input information, design variables and measuring data. The model was trained using a large-scale dataset and showed a good prediction performance. The above three studies also did not consider the evaluation of model response performance.

The optimization framework based on the deep learning surrogate models was presented by Yang et al. [23]. They used a CGAN model along with a genetic algorithm to optimize the hole distribution of effusion cooling under non-uniform incoming flow conditions [23] and to search the optimal shape of pin-fins defined by a high degree spline [28]. The results showed the feasibility and effectiveness of embedding the deep learning-based surrogate models in the optimization framework. However, there was a lack of comparisons of the performance with the conventional surrogate models.

#### Limitation of previous studies

The literature survey showed that most deep learning models were trained on limited samples, and the prediction accuracy of models was only evaluated at several single points, which might lead to overfitting problems. A reliable surrogate model should respond to different design variables properly within a wide range. The sensitivity of output parameters to input variables should also be predicted correctly by surrogate models. Additionally, the model accuracy should be relatively robust to sample sizes. The literature still lacked a comprehensive comparison of the characteristics mentioned above between parameter-based models and image-based models.

#### Objective of this study

This study was dedicated to evaluating the conventional and deep learning-based surrogate modeling methods in predicting the flow and heat transfer performance of U-bend channels with variable geometries. Three conventional (i.e., ANN, RBF and Kriging) and two deep learning-based (i.e., CNN and CGAN) surrogate models were established in this study.

The contribution of this study is threefold: 1) deep learning methods were used to hopefully predict 2-D physical fields; 2) a large-scale dataset containing 3160 samples was generated for model analysis to hopefully improve the reliability of the analysis results, which was unique among the existing studies in the literature; 3) the single-point prediction accuracy, response accuracy, sensitivity to sample size, and other characteristics of interest of these models were analyzed in detail to hopefully make a comprehensive evaluation, while most of the existing studies in the literature only evaluated the single-point prediction performance.

The efforts were expected to reveal the characteristics of different kinds of modeling methods, so as to provide guidance on establishing reliable data-driven surrogate models for the design of thermal energy conversion equipment.

#### Dataset preparation

##### Geometries

**Fig. 1(a)** shows the baseline geometry of U-bend channels investigated by the present study. The hydraulic diameter and aspect ratio of the channel was 0.075 m and 1, respectively. The shape of the U-bend varied within a given bounding box. The degree of freedom of geometric deformation was 26. The parameterization of the outer and inner profile of the U-bend is defined in **Fig. 1(b)** and **(c)**, respectively. Both curves were composed of four Non-Uniform Rational Bzier Splines (NURBS) [29] curves. Each NURBS curve was defined by four control points and therefore was a third-order curve. Consequently, the shape of the U-bend could be controlled by adjusting the coordinates of these control points.

The outer profile was controlled by 13 control points arranged clockwise along the curve, of which the first and last points were fixed reference points. The outer profile had a total of 12 degrees of freedom, as shown by the arrows in the figure. In order to guarantee 1st ordered gradient (G1) continuity, the location of several control points in one

**Table 1**

A summary of design parameters of parameterized U-bends.

Parameter	Range	Parameter	Range
$y - p_2^{out}$	[0.3, 0.7]	$y - p_3^{in}$	[0.2, 0.5]
$y - p_3^{out}$	[0.2, 0.6]	$y - p_4^{in}$	[1.1, 1.6]
$x - p_4^{out}$	[-0.14, -0.08]	$d1$	[0.90, 1.05]
$y - p_4^{out}$	[1.4, 2.0]	$y - p_5^{in}$	[0.1, 0.5]
$x - p_6^{out}$	[-0.04, 1.0]	$x - p_6^{in}$	[-0.15, 0.20]
$x - p_7^{out}$	[0, 0.8]	$x - p_7^{in}$	[0, 0.15]
$x - p_8^{out}$	[-0.30, 0.15]	$d2$	[1.1, 1.2]
$y - p_9^{out}$	[0.60, 0.85]	$x - p_8^{in}$	[0, 0.25]
$x - p_{10}^{out}$	[0.25, 0.40]	$y - p_9^{in}$	[0.4, 0.6]
$y - p_{10}^{out}$	[2.0, 2.4]	$d3$	[1.05, 1.25]
$y - p_{11}^{out}$	[0.60, 1.10]	$y - p_{10}^{in}$	[1.4, 1.6]
$y - p_{12}^{out}$	[0.4, 0.8]	$y - p_{11}^{in}$	[0.3, 0.8]
$y - p_2^{in}$	[0.3, 0.5]	$y - p_{12}^{in}$	[0.3, 0.5]

direction was determined by the positions of a neighboring control point. In addition, the fifth control point had no degree of freedom, and its position was determined by the third, fourth, and sixth control points of the outer profile to guarantee 2st ordered gradient (G2) continuity. The parameterization of the inner profile was similar to that of the outer profile. The inner profile was parametrized by 14 parameters, among

which,  $d1$ ,  $d2$  and  $d3$  were defined based on the parameters of the outer profile. More details about parameterization could be found in the work of Verstraete et al. [11]. Therefore, the total number of parameters for the entire U-bend was 26. Table 1 summarizes these 26 parameters and their range of values. All parameters were nondimensionalized by the hydraulic diameter.

The Latin Hypercube Sampling (LHS) method [30] was used to generate the parameter matrix. A sufficiently large database was considered to be generated to better study the performance of different surrogate models. Therefore, 2000, 400 and 400 design points were generated within the parameter range respectively for model training, validation and testing. It should be noted that 400 samples were more than sufficient for model validation or testing for engineering applications where a large-scale dataset could not be generated due to the high cost of experiments and numerical calculations. The design points generated by the LHS method randomly covered the design space with the magnitude of each parameter that occurred uniformly. This procedure was important for neural network training because it tended to push the samples closer to the independent identically distribution.

#### Computational setups

This study utilized the automatic modeling-meshing-simulation-post processing function of the ANSYS workbench platform to perform batch calculations for all design points. A CFX solver with Reynolds Averaged

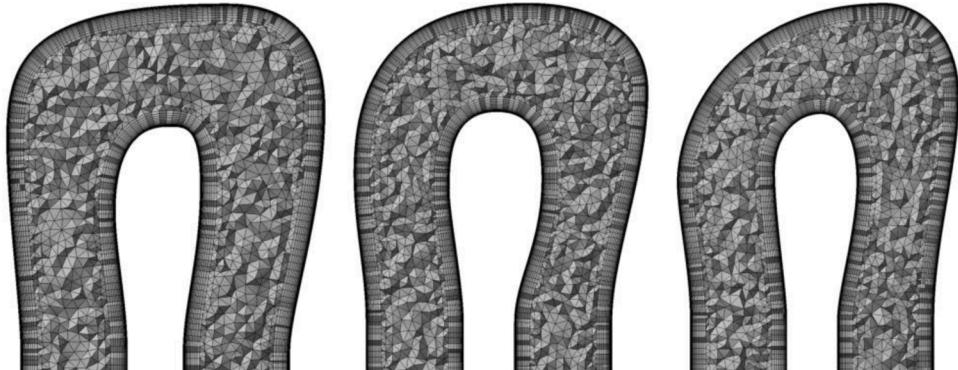


Fig. 2. Mesh local views of three U-bends with different values of design parameters.

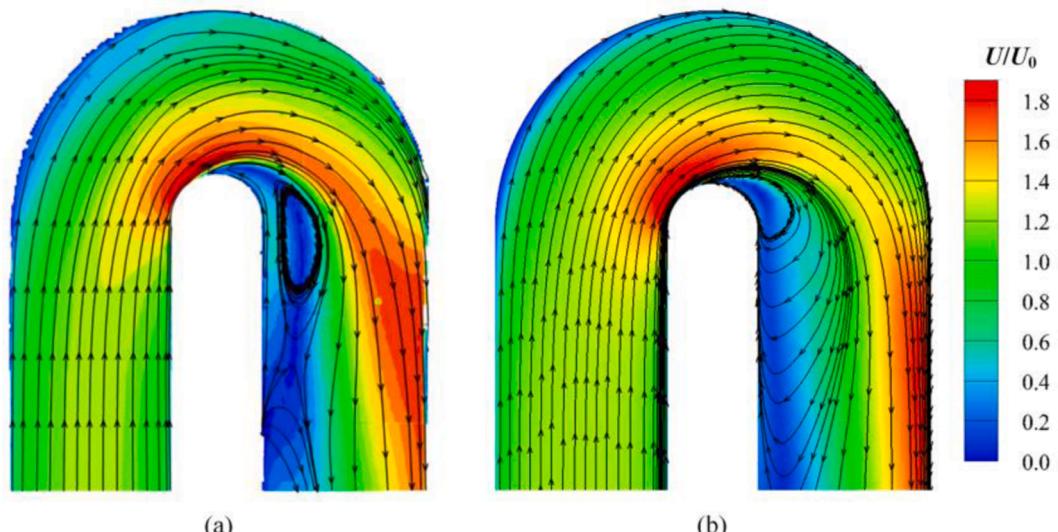


Fig. 3. The distributions of normalized velocity and streamlines at the middle section of the U-bend. (a) experimental result published by Coletti et al. [12]; (b) numerical result obtained in this study.

**Table 2**

A summary of the dataset generated in this study.

Data	Shape
Design parameters	$2800 \times 26$
Pressure drop ( $\Delta P$ )	$2800 \times 1$
Heat transfer ( $Q$ )	$2800 \times 1$
Pressure distribution ( $P$ )	$2800 \times 512 \times 256 \times 1$
Heat flux distribution ( $q$ )	$2800 \times 512 \times 256 \times 1$

Navier-Stokes (RANS) models was employed to predict the flow and heat transfer performance. The Shear Stress Transport (SST)  $k-\omega$  model was adopted for turbulence closure. A static temperature of 293.15 K and a velocity of 8.4 m/s were specified at the inlet. An average static pressure of 101,325 Pa was provided at the outlet. All walls were specified as non-slip wall boundaries with a temperature of 343.15 K. The Reynolds number was 40,000, and the Mach number was less than 0.3, so an incompressible assumption was valid. Fig. 2 shows the local views of meshes which were mainly constructed by tetrahedral elements. The  $y^+$  was guaranteed to be less than 1 globally.

To achieve mesh independence, testing simulations were conducted under three mesh density levels, which were 1.15, 2.64, and 4.63 million, respectively. The normalized static pressure drop ( $\Delta P^*$ ) was selected as the response variable. The  $\Delta P^*$  is defined in Eq. (1) where  $P_i$  and  $P_o$  are the static pressure measured at a distance of  $5D_h$  and  $11D_h$  from the tip of the bend,  $\rho$  is the density of air, and  $U_0$  is the bulk flow velocity. Using the result of the 4.63 million mesh as a reference, the Root Mean Square Error (RMSE) of  $\Delta P^*$  for the other two mesh density levels were 4.8%, and 1.1%, respectively. Thus, the 2.64 million mesh density level was considered to meet the requirement of mesh independence and the corresponding meshing parameters were proper for all design points.

$$\Delta P^* = \frac{P_i - P_o}{\frac{1}{2}\rho U_0^2} \quad (1)$$

To ensure the accuracy of the numerical calculations for the entire batch of design points, the simulation results were compared with the corresponding experimental data published by Coletti et al. [12]. The experimental data of  $\Delta P^*$  was 1.03 while the simulated data was. The relative error was 8.7%, which shows an acceptable uncertainty. Fig. 3 shows the normalized velocity distribution in the U-bend region along the symmetry plane obtained by experiment and numerical calculation. It was found that both the overall flow pattern and the lateral extent of the flow separation were well predicted by the present numerical calculation. Thus, the numerical method was reasonably validated.

### Data post-processing

The U-bend optimization was usually driven by the minimization of pressure drop ( $\Delta P$ ) and the maximization of heat transfer ( $Q$ ).  $\Delta P$  is defined in Eq. (2) where  $P_{up}$  and  $P_{down}$  are the static pressure at the user-defined upstream and downstream sections located at a distance of  $8D_h$  from the tip of the bend.  $Q$  is defined in Eq. (3) where  $q$  is the wall heat flux, and  $A$  is the region enclosed by the user-defined upstream and downstream positions. The  $\Delta P$  and  $Q$  of all design points were calculated and used for conventional surrogate modeling. The corresponding pressure distribution and heat flux distribution of all design points were post-processed into  $512 \times 256 \times 1$  images and used for deep learning-based surrogate modeling.

$$\Delta P = P_{up} - P_{down} \quad (2)$$

$$Q = \int_{tip} q dA \quad (3)$$

Table 2 provides a summary of the generated dataset. There were a total of 2800 samples, including 2000 training samples, 400 validation samples, and 400 test samples, each of which had five attributes. It took around  $5.0 \times 10^4$  CPU-hours for all numerical calculations. The mapping between the design parameters and the physical information was established by the surrogate modeling methods described in the next section.

## Methodology

### Surrogate modeling ideas

The surrogate model aimed to establish a mapping between design parameters and objective values to achieve fast prediction of pressure drop and heat transfer amount. Fig. 4 shows three modeling methods used in this study: 1) The first category of models directly established a mapping between design parameters and objective values. The models included ANN, RBF and Kriging; 2) The second category of models are parameter-image mappings established between design parameters and the flow/temperature fields, while objective values were further calculated by integrating the fields. The selected model was CNN; 3) The third category of models are image-image mappings established between the geometric profile and the flow/temperature fields. The selected model was CGAN. This study aimed to comprehensively evaluate the three modeling methods and the five surrogate models to provide guidance for developing a reliable and practical data-driven surrogate model.

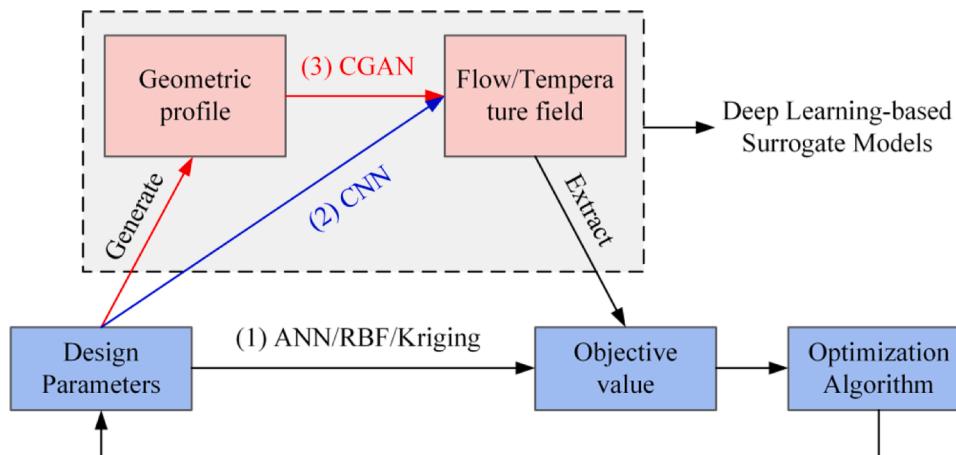
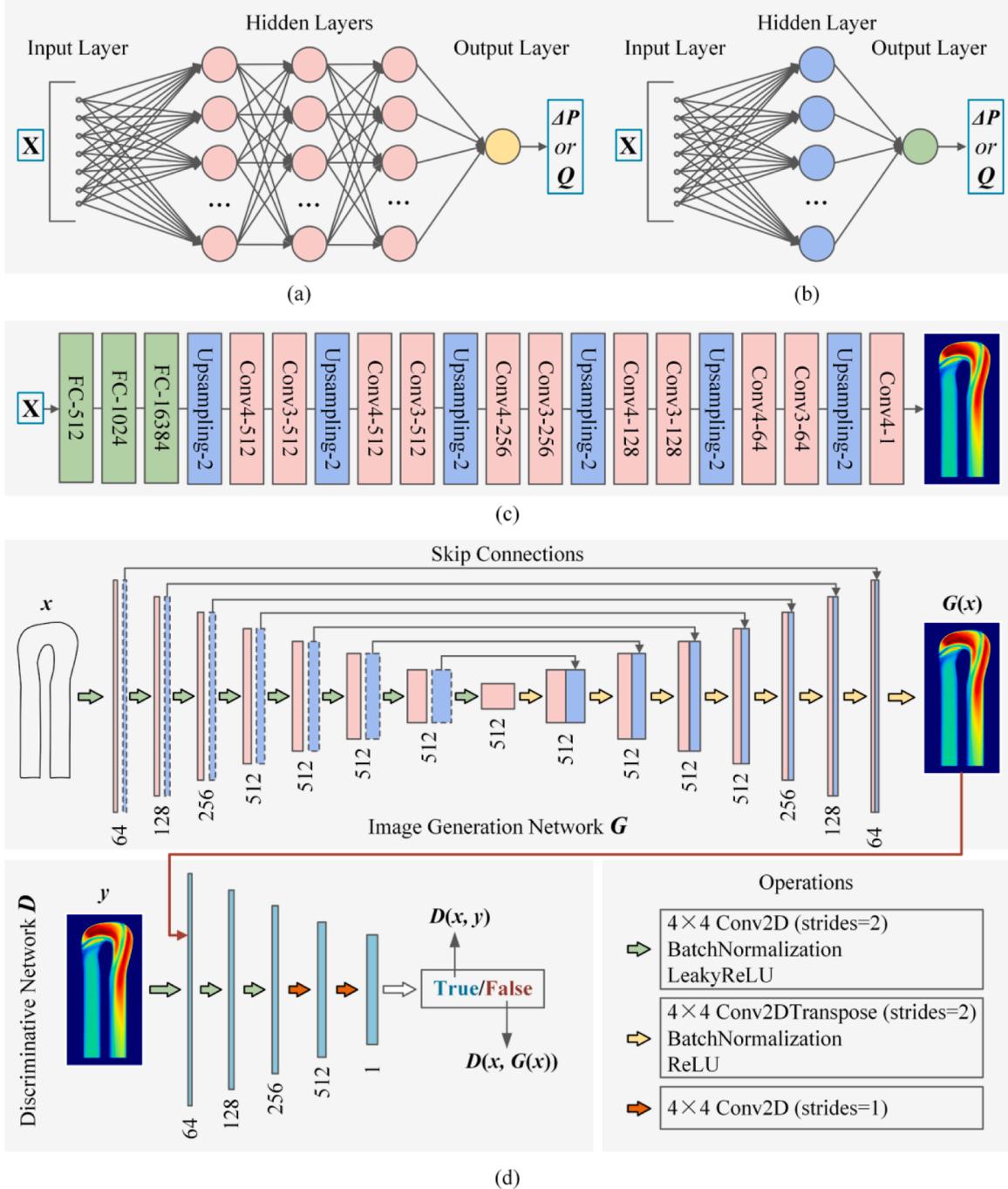


Fig. 4. Three different types of surrogate modeling ideas.



**Fig. 5.** The architecture of different neural network models established in this study. (a) ANN; (b) RBF; (c) CNN; (d) CGAN.

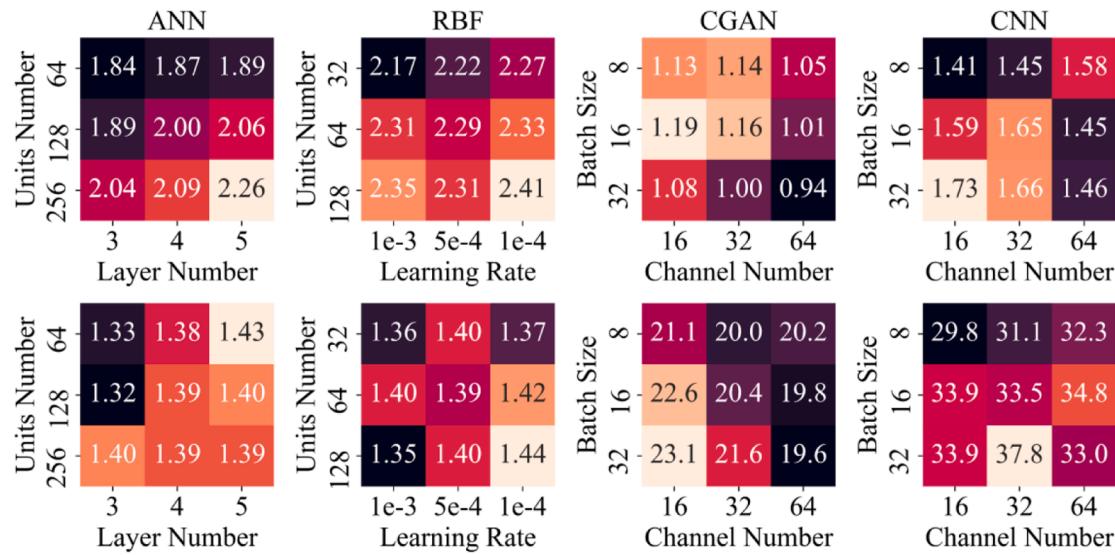
#### Model architectures

The architecture of the ANN model is shown in Fig. 5(a), which consists of an input layer,  $N$  hidden layers and an output layer. Each layer had several neurons. The input layer received a vector  $\mathbf{X}$  composed of 26 design parameters, and the output layer was the pressure drop or heat transfer. The outputs of the  $j$ th neuron in the first and  $l$ th layers are given by Eq. (4) and (5), respectively. Among them,  $x$  represents the input feature,  $a$  represents the output of each layer,  $\sigma$  represents a nonlinear activation function, generally Rectified Linear Unit (ReLU),  $w_{jk}$  represents the weight of the connection between the  $j$ th neuron of the current layer and the  $k$ th neuron of the previous layer, and  $b$  represents the bias.

$$a_j^l = \sigma_l \left( \sum_{k=1}^{26} w_{jk}^l x_k + b_j^l \right) \quad (4)$$

$$a_j^l = \sigma_l \left( \sum_{k=1}^{N^{(l-1)}} w_{jk}^l a_k^{l-1} + b_j^l \right) \quad (5)$$

The RBF model shown in Fig. 5(b) had only one hidden layer, neurons of which is activated by radial basis functions. The Gaussian kernel function given by Eq. (6) is a typical radial basis function. Among them,  $x$  is the input feature,  $c_j$  is the center point of the  $j$ th neuron, and  $\sigma$  is the width of the Gaussian kernel. The output layer was not activated so that the final output of the network was the linear weighted summation of the output of the hidden layer units. Consequently, the RBF model



**Fig. 6.** The results of hyperparameter tuning for each neural network model.

greatly reduced the number of parameters while ensuring that the network had a certain nonlinear mapping capability, thereby reducing the order of the model and speeding up the learning speed. Eq. (7) shows the mathematical expression of the RBF model, where  $\omega_j$  is the weight of the  $j$ th neuron.

$$\phi(x_i, c_j) = e^{-\frac{\|x_i - c_j\|_2}{2\sigma^2}} \quad (6)$$

$$f(x) = \sum_{j=1}^k w_j \cdot \phi(x_i, c_j) \quad (7)$$

The Kriging model was an interpolation model and belonged to the series of linear least-squares algorithms. The mathematical expression of the standard Kriging model consists of two parts, as shown in Eq. (8). The first part is a linear regression term used to capture the overall trend of the response. Among them,  $g(x)$  is the regression function, and  $\beta$  is the weight coefficient. The second part  $Z(x)$  is used to describe the random process of the difference between the overall trend and the local trend of the response.  $Z(x)$  is a model of a Gaussian stationary random process with zero means. The parameter  $\beta$  and the function  $Z(x)$  are usually determined in such a way that  $f(x)$  is the best linear unbiased estimator. Therefore, the Kriging model could obtain the best linear unbiased estimate of the observed value.

$$f(x) = \sum_{j=1}^k \beta_j g_j(x) + Z(x) \quad (8)$$

Fig. 5(c) shows the architecture of the CNN model, which was proposed on the backbone of the image generator developed by Dosovitskiy et al. [31]. The input of the model was the 26 design parameters, and the output was the pressure or heat flux distributions. The parameter vector  $X$  was firstly encoded into a vector of length 16,384 by three fully connected layers and then reshaped into an  $8 \times 4 \times 512$  feature map. This feature map was converted to a  $512 \times 256 \times 1$  single-channel image after six up sampling and convolution operations. The output layer was activated by the tanh function, while the other layers were activated by the ReLU function. The CNN model proposed in the present study established the mapping between the low-dimensional parameter vector and the high-dimensional physical quantity distribution, which provided convenience for parameter optimizations.

Fig. 5(d) shows the architecture of the CGAN model, which was proposed on the backbone of the pix2pix network developed by Isola et al. [32]. The input was the geometric profile, and the output was the

pressure or heat flux distributions. The CGAN model consisted of a generator ( $G$ ) and a discriminator ( $D$ ), which played an adversarial role against each other.  $G$  was an encoder-decoder with the U-net as the backbone. Skip connections drawn with dashed lines in Fig. 5(d) were established between the layers with identical feature map sizes to enable  $G$  to accurately identify the boundaries of the physical field.  $D$  was a CNN-based encoder whose function was to recognize the fake data  $G(x)$ . After a continuous confrontation, a balance was reached between  $G$  and  $D$ . In the end, the prediction result of the CGAN model (i.e.,  $G(x)$ ) was very similar to the ground truth (i.e., CFD data).

#### Implementation details

The five surrogate models were all coded in Python. Except for the Kriging model which was implemented using the pyKriging open-source library, the other four neural network models were all implemented using the TensorFlow open-source framework. All the training samples were pre-shuffled before being fed into the neural network model. The trainable parameters of all neural network models were optimized by the adaptive moment estimation (Adam) optimizer. The training of the CNN and CGAN models was accelerated by an NVIDIA Quadro RTX 5000 GPU processor.

The hyperparameters of four neural network models were carefully selected. For the ANN model, the number of hidden layers and the number of neurons in each hidden layer were selected as the adjustable hyperparameters, while the learning rate and batch size were fixed to 2e-4 and 32, respectively. For the RBF model, the learning rate and the number of neurons in the hidden layer were adjustable, and the batch size was fixed to 32. The adjustable hyperparameters of the CNN and CGAN models were the number of basic channels and batch size, and their learning rates were fixedly set to 5e-4 and 2e-4, respectively. The optimal hyperparameter combination was searched using a  $3 \times 3$  grid, and the accuracy of the model under each hyperparameter combination was evaluated on the validation set containing 400 samples.

Fig. 6 shows the results of hyperparameter tuning for each neural network model. The subfigures in the first and second rows represent the results when predicting flow (i.e.,  $\Delta P$  or  $P$ ) and heat transfer performance (i.e.,  $Q$  or  $q$ ), respectively. The numbers in each subfigure represent prediction errors. For the ANN and RBF model, the error was the RMSE between the predicted value and the ground truth value. For the CNN and CGAN model, the error was the pixel-wise Mean Absolute Error (MAE) between the predicted image and the ground truth image. The RMSE and MAE are defined in Eqs. (9) and (10), respectively.

**Table 3**

The optimal hyperparameter combination for each neural network model.

Model	Hyperparameter	Value for predicting $\Delta P$ or $P$	Value for predicting $Q$ or $q$
ANN	Number of hidden layers	3	3
	Number of units per hidden layer	64	64
RBF	Learning Rate	1e-3	1e-3
	Number of units per hidden layer	32	128
CNN	Number of basic channels	16	16
	Batch size	8	8
CGAN	Number of basic channels	64	64
	Batch size	32	32

**Table 3** summarizes the obtained optimal hyperparameter combinations for each model. The prediction results of each model shown in subsequent sections were obtained under the optimal hyperparameter combinations.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (9)$$

$$\text{MAE} = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (10)$$

## Results and discussion

### Single-point accuracy

#### Conventional models

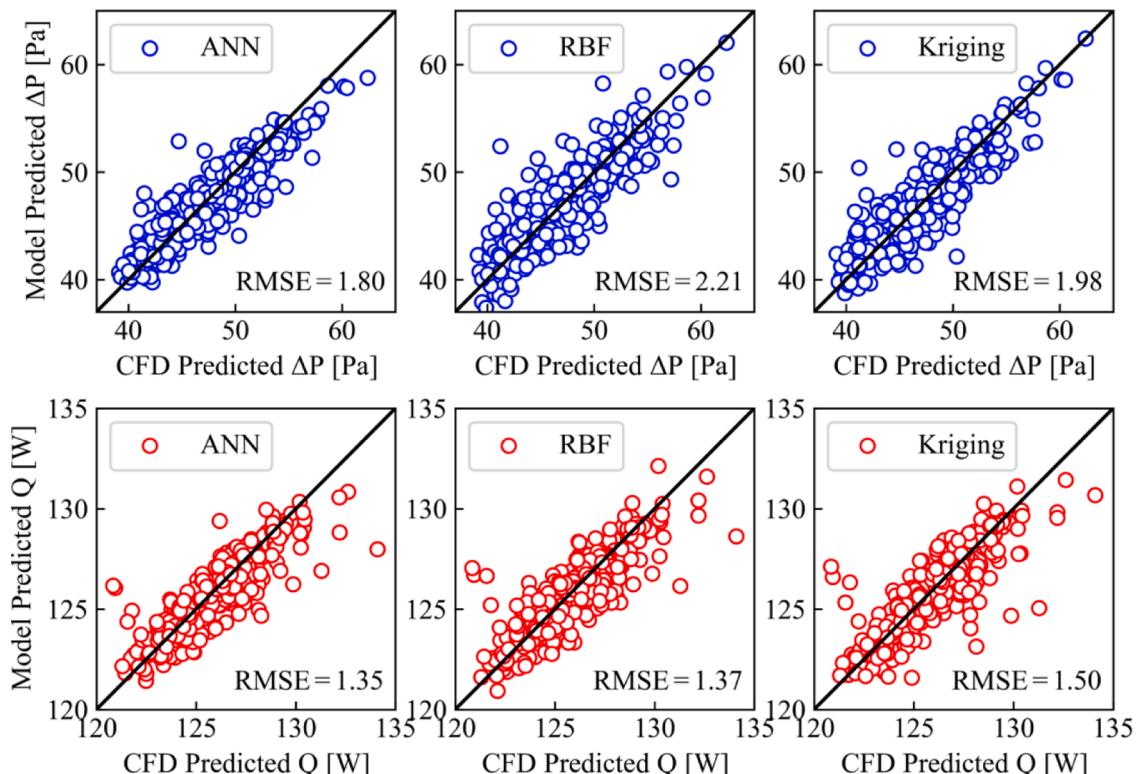
Fig. 7 shows the prediction accuracy at each single sampling point of the three conventional models. The subfigures in the first row and the second row represent the comparison of the prediction results of the  $\Delta P$  and  $Q$  with the CFD results, respectively. For the prediction of the  $\Delta P$ , the RMSE values of the ANN, RBF and Kriging model were 1.80 Pa, 2.21 Pa and 1.98 Pa, and the corresponding relative errors were  $\pm 6.0\%$ ,  $\pm 7.4\%$ , and  $\pm 6.6\%$ , respectively. For the prediction of the  $Q$ , the RMSE of the ANN, RBF and Kriging model was 1.35 W, 1.37 W and 1.50 W, and the corresponding relative errors were  $\pm 5.6\%$ ,  $\pm 5.7\%$ , and  $\pm 6.3\%$ , respectively. It could be found that all the three surrogate models had good prediction performance. Compared with the RBF model and the Kriging model, the ANN model with stronger high-dimensional and non-linear fitting ability could learn from data more effectively in the case of large sample size (i.e., 2000).

the RMSE of the ANN, RBF and Kriging model was 1.80 Pa, 2.21 Pa and 1.98 Pa, and the corresponding relative errors were  $\pm 6.0\%$ ,  $\pm 7.4\%$ , and  $\pm 6.6\%$ , respectively. For the prediction of the  $Q$ , the RMSE of the ANN, RBF and Kriging model was 1.35 W, 1.37 W and 1.50 W, and the corresponding relative errors were  $\pm 5.6\%$ ,  $\pm 5.7\%$ , and  $\pm 6.3\%$ , respectively. It could be found that all the three surrogate models had good prediction performance. Compared with the RBF model and the Kriging model, the ANN model with stronger high-dimensional and non-linear fitting ability could learn from data more effectively in the case of large sample size (i.e., 2000).

#### Deep learning-based models

Figs. 8 and 9 show the ability of the deep learning-based models to predict high-dimensional physical information under different geometric conditions. Five cases were selected to visualize the prediction accuracy of flow/temperature fields on the test set in detail. For each case, the U-bend profile and the results of the pressure distribution and wall heat flux distribution predicted by CFD and deep learning models were plotted. The subfigure on the right of each model prediction shows the distribution of five times pixel-wise absolute error. The results showed that both deep learning models could well reconstruct the complex flow and heat transfer distribution. Although the CGAN model used different input dimensions from CNN (i.e., parameters against image), both models captured the geometry profile properly. For the pressure distribution, the absolute error was mostly within 0.05 and mainly distributed on the second channel. For the wall heat flux distribution, the absolute error was mostly within 0.1 and mainly distributed on the bend. Overall, the prediction accuracy of the CGAN model was slightly higher than that of the CNN model. In addition, the accuracy of both deep learning models in predicting the pressure distribution was higher than predicting the wall heat flux distribution. One possible reason for this was that the convective heat transfer introduced additional nonlinearities.

More quantitatively, the probability distribution of the pixel-wise Mean Absolute Error (MAE) of both deep learning models on the



**Fig. 7.** The accuracy of conventional models to predict the pressure loss and heat transfer on the entire test set.

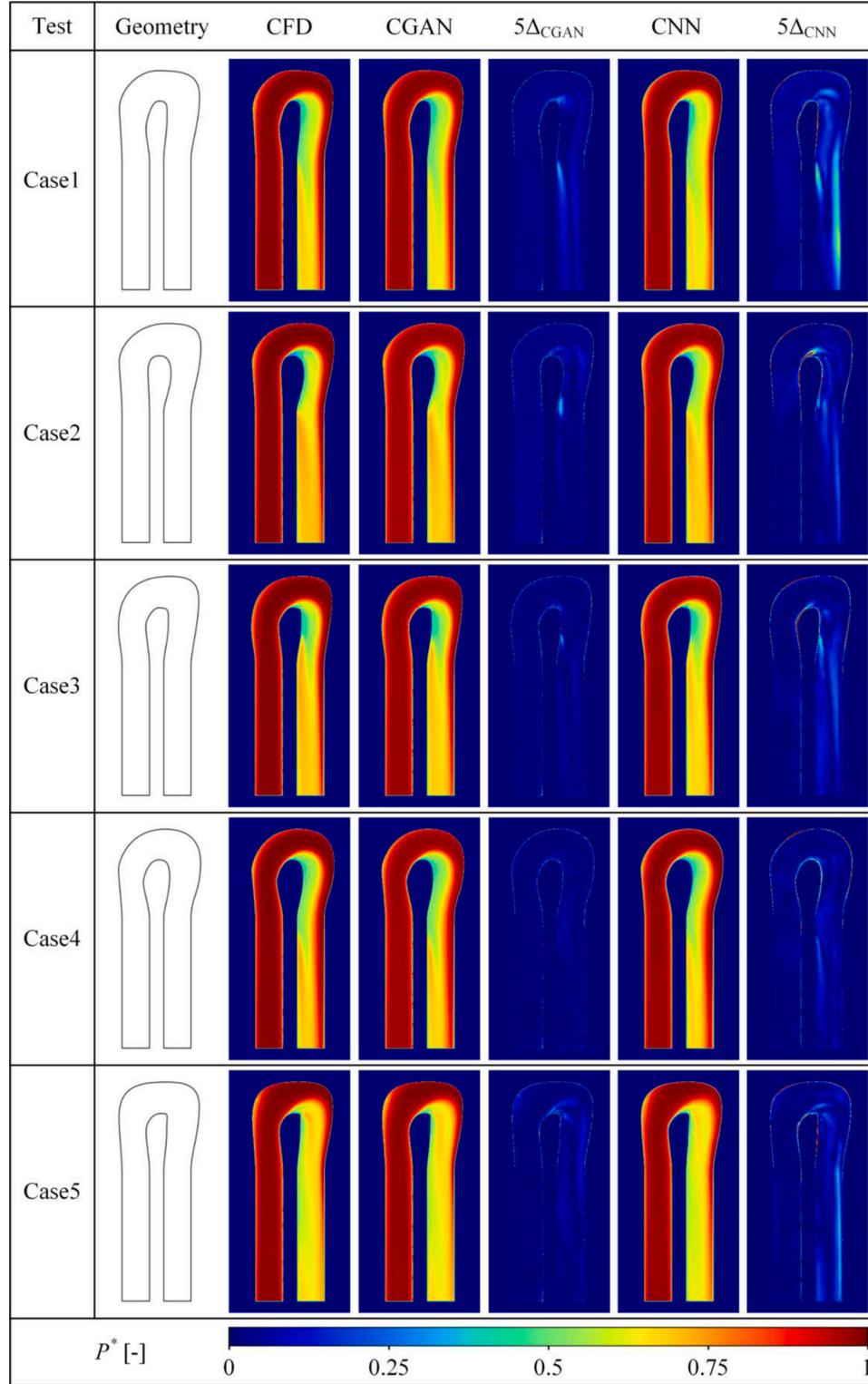


Fig. 8. The accuracy of deep learning-based models to predict the pressure distribution.

entire test set is presented in Fig. 10. For the pressure distribution, the pixel-wise MAE of the CGAN model was more focused within the range of 2.0 Pa, with an average value of 0.95 Pa. The CNN model had a larger portion of pixel-wise MAE beyond 2.0 Pa. The average pixel-wise MAE of the CNN model was 1.41 Pa, which was 48% higher than that of the CGAN model. Overall, the pixel-wise MAE of both deep learning models was almost within the range of 6.0 Pa, which was only 3.2% of the pressure variation range. For the wall heat flux distribution, the pixel-

wise MAE of the CGAN model was more focused within the range of 45 W/m<sup>2</sup>, with an average value of 20.3 W/m<sup>2</sup>. The CNN model had a larger portion of pixel-wise MAE beyond 45 W/m<sup>2</sup>. The average pixel-wise MAE of the CNN model was 29.8 W/m<sup>2</sup>, which was 47% higher than that of the CGAN model. Overall, the pixel-wise MAE of both deep learning models were almost within the range of 135 W/m<sup>2</sup>, which was only 3.1% of the wall heat flux variation range. Therefore, both deep learning models could accurately predict the pressure distribution and

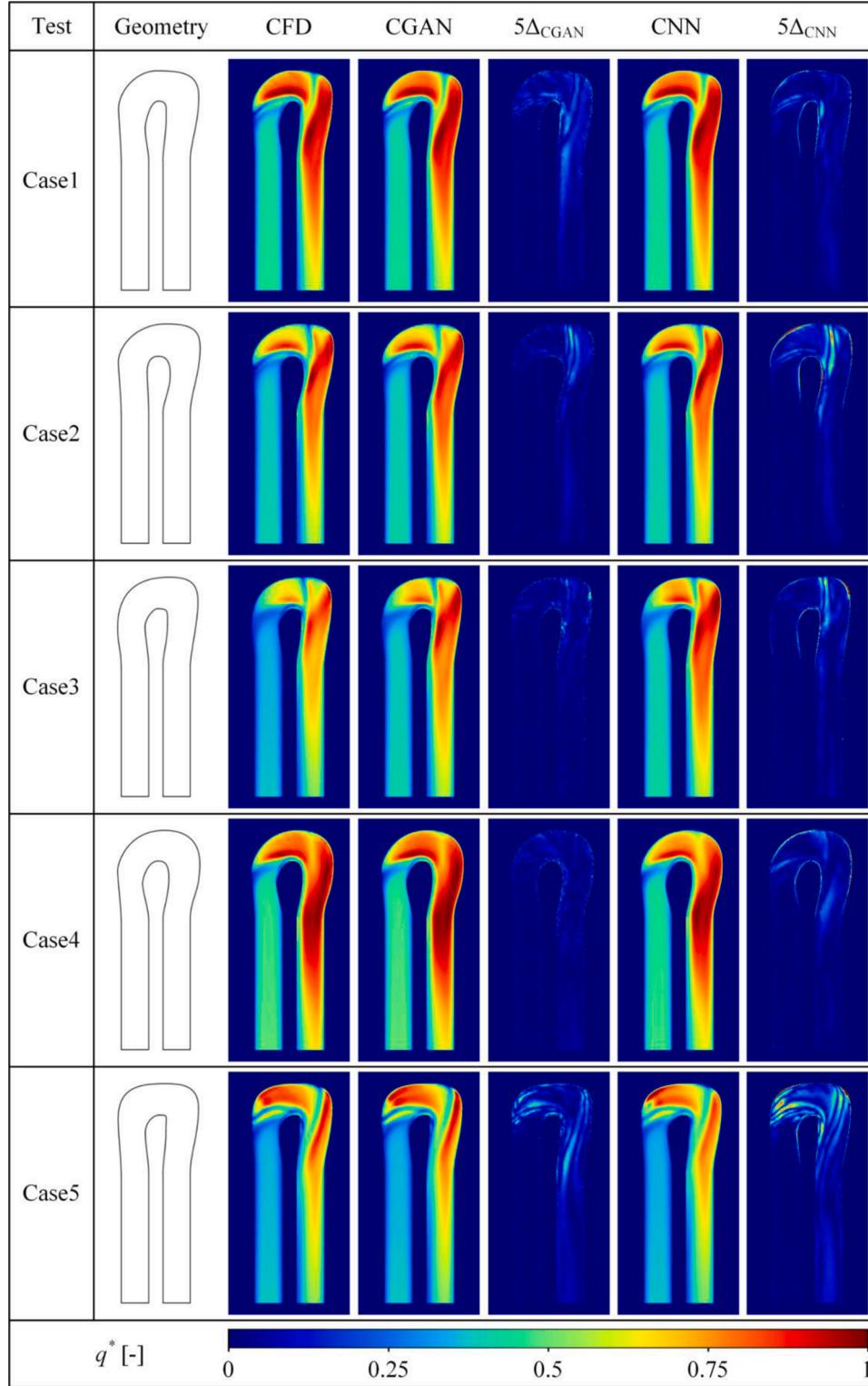


Fig. 9. The accuracy of deep learning-based models to predict the wall heat flux distribution.

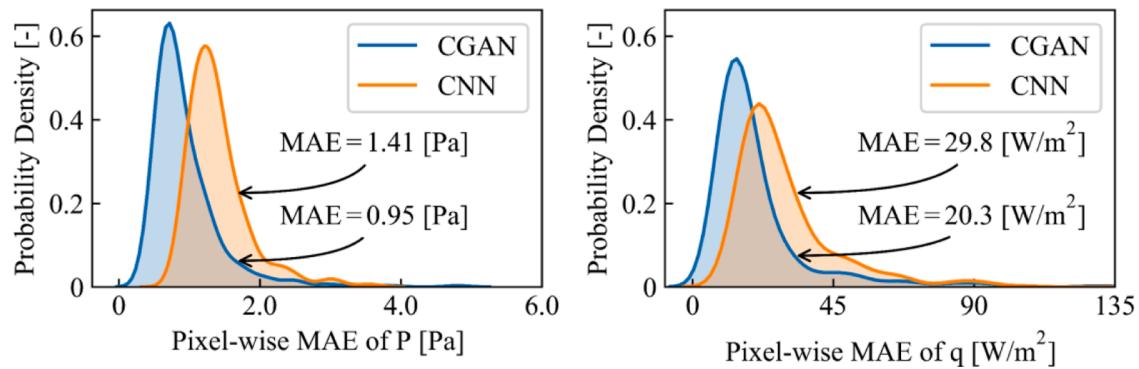
the wall heat flux distribution of the U-bend channel with variable geometries, and the CGAN model that directly took geometric profiles as input performed better.

The corresponding objective values (i.e.,  $\Delta P$  and  $Q$ ) could be easily extracted from the predicted distributions. The subfigures in the first row and the second row of Fig. 11 show the comparison of the prediction results of the  $\Delta P$  and  $Q$  with the CFD results, respectively. The results showed that the CGAN and CNN model had close prediction accuracy.

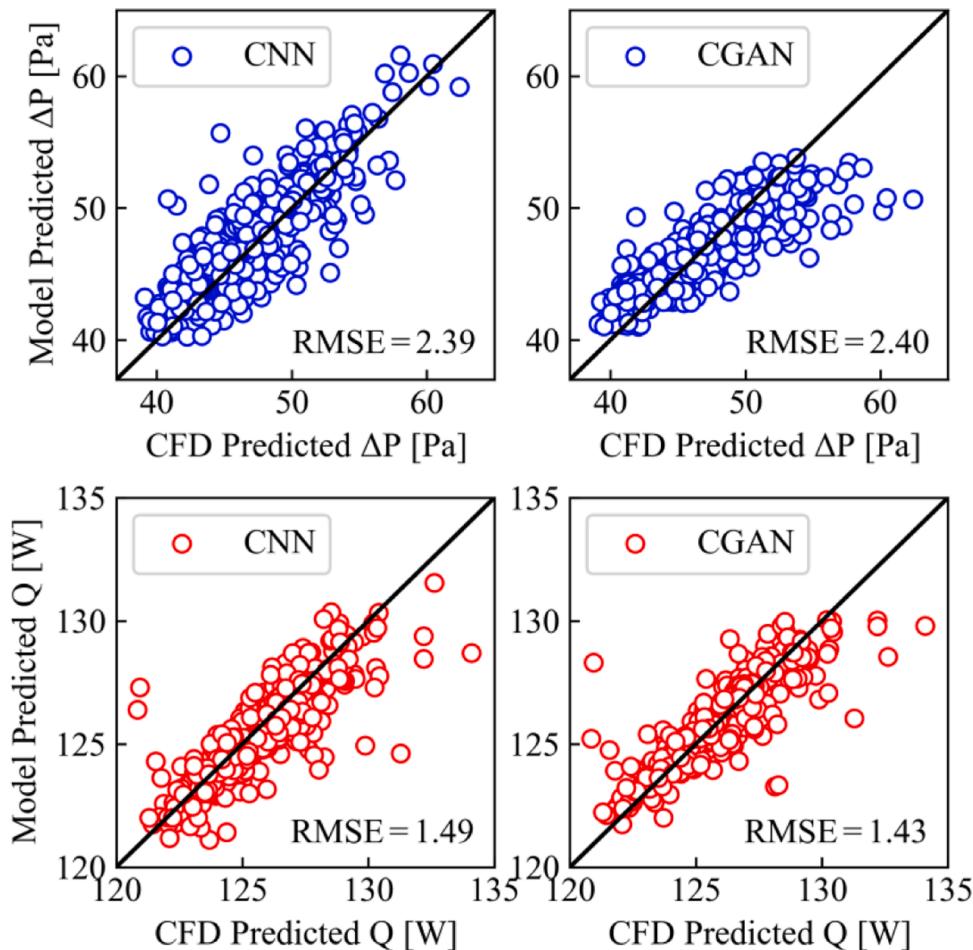
The RMSE of the  $\Delta P$  and  $Q$  was close to 2.40 Pa and 1.50 W, and the corresponding relative error was about  $\pm 8.0\%$  and  $\pm 6.3\%$ , respectively. Therefore, both deep learning models could also accurately predict the  $\Delta P$  and  $Q$ , which indicated that they both had the potential to be reliable surrogate models.

#### Quantitative comparison

Fig. 12 quantitatively compares the prediction accuracy of different



**Fig. 10.** The probability distributions of the pixel-wise absolute prediction error of deep learning-based models on the entire test set.



**Fig. 11.** The accuracy of deep learning-based models to predict the pressure loss and heat transfer on the entire test set.

models. The models with the highest prediction accuracy in the conventional and deep learning-based methods are marked by solid and hollow five-pointed stars, respectively. It was found that the ANN model performed best among the five models whether in predicting the  $\Delta P$  or  $Q$ . The RMSE of the deep learning models were 32.8% and 5.9% higher than that of the ANN model in predicting the  $\Delta P$  and  $Q$ , respectively. Although the prediction accuracy of the deep learning models was a little lower than that of the conventional models, they were still with low and acceptable prediction errors.

#### Response performance

Although the prediction accuracy of all surrogate models had been verified on a relatively large enough test set with 400 samples in the previous section, data-driven models inevitably have the risk of overfitting. The reason for this suspect was due to the fact that data-driven models, especially deep learning-based models, usually had tens of thousands to tens of millions of parameters, which was far more than the order of the studied problem and the number of training samples. Therefore, it was important to adequately evaluate the generalization capability of the data-driven models in a wider parameter range to ensure good accuracy in practical design and optimization. An effective

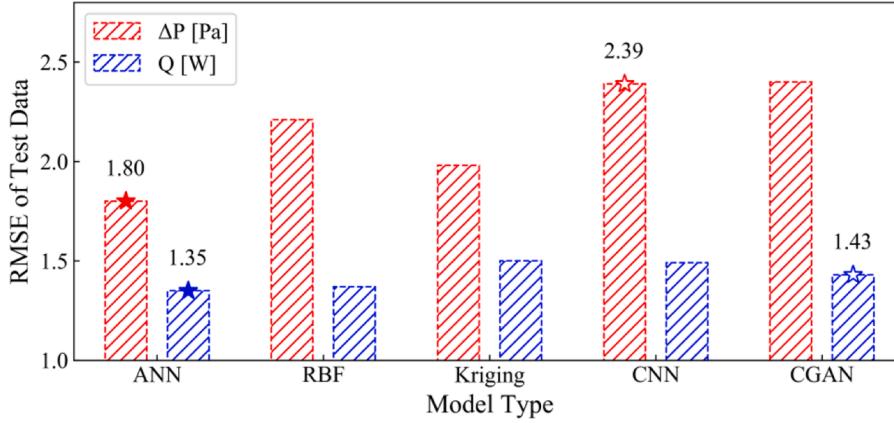


Fig. 12. A quantitative comparison of the prediction accuracy of different models.

way to achieve this goal was to conduct parameter response analyses. By analyzing the variations of outputs by continuously changing the inputs, the generalization accuracy of the model could be verified more rigorously. The response of the model to the input parameters should be consistent with that of the CFD simulations if the model was properly fitted.

The response analysis in the present study was conducted for some parameters that had a greater impact on the shape of the U-bend, including five outer curve parameters ( $y_{p_3}^{out}$ ,  $y_{p_4}^{out}$ ,  $x_{p_7}^{out}$ ,  $x_{p_{10}}^{out}$ ,  $y_{p_{10}}^{out}$ ) and five inner curve parameters ( $y_{p_4}^{in}$ ,  $x_{p_6}^{in}$ ,  $d_1$ ,  $d_2$ ,  $d_3$ ). Six groups of responses were performed for the U-bend channel, of which the first five groups were for a single parameter, and the last group was for a combination of five parameters. Each selected parameter was evenly divided into 30 equal parts within its value range. Thus, there were 360 points in the response dataset. The center point of the response analysis was at the median value of all parameters. The five trained surrogate models were used to predict the  $\Delta P$  and  $Q$  of all design points in the response set to verify the consistency of the parameter response by comparing against the CFD response. Figs. 13 and 14 show the quantitative comparison between the five surrogate models and the CFD in the response of  $\Delta P$  and  $Q$  to different geometric parameters, respectively. Each subfigure represented the response to one variable while keeping the other parameters consistent with the center point. In all subfigures, the response results of the five surrogate models and the CFD are drawn by five different colored dashed lines and red bold solid lines, respectively.

For the response of  $\Delta P$  shown in Fig. 13, the results of the ANN model and the CGAN model were in good agreement with the CFD results no matter in the trend or the magnitude, with the RMSE of 0.67 Pa and 0.57 Pa, respectively. The CNN model and the Kriging model overestimated and underestimated the  $\Delta P$ , and with the RMSE of 1.57 Pa and 0.86 Pa, respectively. The response performance of the RBF model was between the CNN model and the Kriging model, which had an RMSE of 1.24 Pa. Overall, the maximum average relative error in predicting the  $\Delta P$  on the response dataset of the five surrogate models was  $\pm 5.2\%$ , which was lower than that on the test dataset. This further proved that the five surrogate models trained in this study all could accurately predict the pressure loss. Meanwhile, compared with the other three models, the ANN model and the CGAN model not only had higher generalization accuracy, but also had higher response accuracy, which indicated that they could better model the inherent physical laws between pressure loss and geometric parameters.

For the response of  $Q$  shown in Fig. 14, the results of the ANN, Kriging, CNN and CGAN model agreed well with the CFD results both in the trend or the magnitude, with the RMSE of 0.37 W, 0.33 W, 0.26 W, and 0.29 W, respectively. The maximum average relative error of these four models on the response dataset was  $\pm 1.5\%$ , which was significantly

lower than that on the test dataset. The RBF model underestimated the response results obviously and given inconsistent trends, with the RMSE of 2.0 W. The average relative error of the RBF model on the response dataset was  $\pm 8.3\%$ , which was higher than that on the test dataset. Except for the RBF model, the generalization accuracy of the other four surrogate models to predict the heat transfer had been further verified. Meanwhile, these four surrogate models could also respond reasonably, which indicated that they could better model the inherent physical laws between heat transfer and geometric parameters.

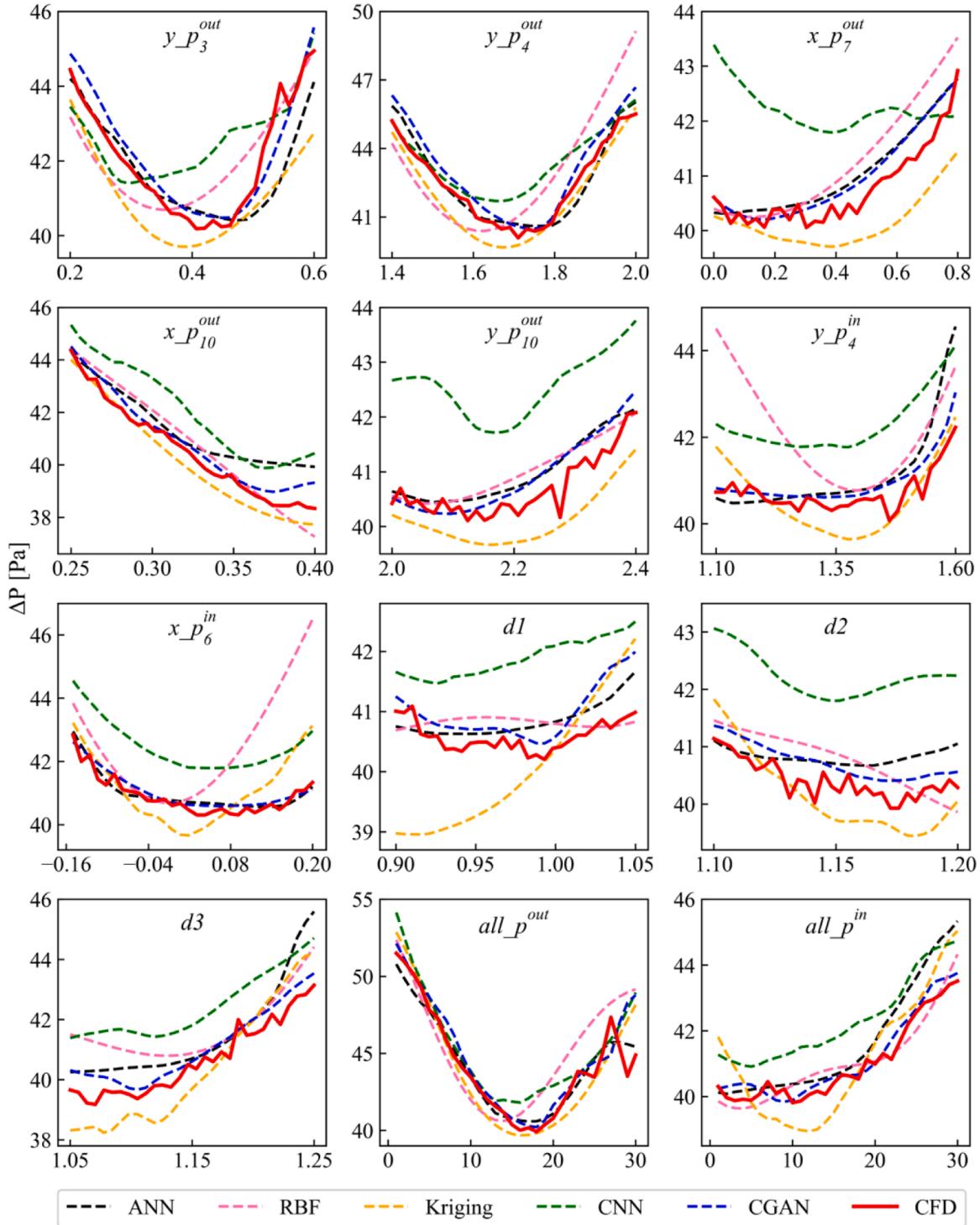
Considering the fact that the consistency of response trends between models and CFD predictions was more important than the matching of the magnitudes, the present study utilized the Pearson correlation coefficient defined in Eq. (11) to more reasonably evaluate the response performance of different models. The  $x$  and  $y$  in Eq. (11) represent the model prediction result and the CFD result, respectively. Fig. 15 shows the quantitative comparison results. The models with the highest correlation coefficient in the conventional and deep learning-based methods are marked by hollow and solid five-pointed stars, respectively.

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (11)$$

It was found that the response results of most models were highly correlated with that of CFD except for that of the RBF model. The CGAN model performed best among the five models whether for the response of the  $\Delta P$  or  $Q$ , and the corresponding Pearson correlation coefficients were 0.97 and 0.96, respectively. The ANN model and Kriging model were the best conventional surrogate models for the response of the  $\Delta P$  or  $Q$ , and the corresponding Pearson correlation coefficients were 0.94 and 0.95, respectively, which were very close to the value of the CGAN model. The RBF model had the worst performance among all models, and the corresponding Pearson correlation coefficients were 0.85 and 0.86, respectively. The results indicated that the modeling method that directly learns the mapping between the geometric profile and the physical information distribution could better mine the physical laws contained in the data.

#### Sensitivity to sample size

The accuracy of data-driven models was usually sample size-dependent. The model could not be adequately trained if the sample size was small, which could lead to poor generalization accuracy. However, due to the limitations of CFD and experimental costs, it was hard to obtain large-scale training samples (above  $10^3$ ) in many engineering fields. Consequently, it was necessary to study the sensitivity of the accuracy of different data-driven models to sample size, so as to

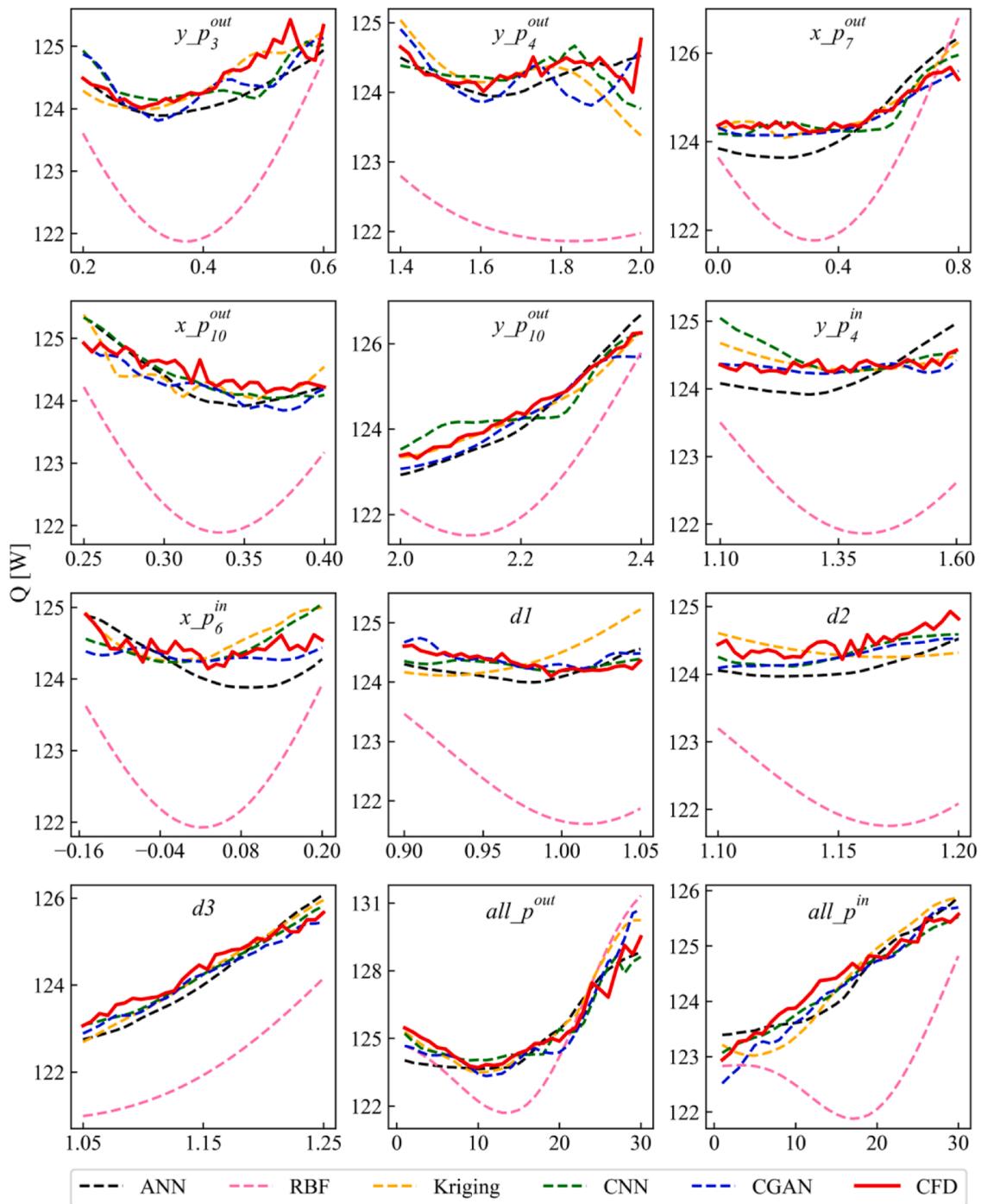


**Fig. 13.** A quantitative comparison between the different surrogate models and CFD in the response of  $\Delta P$  to different parameters.

provide guidance for establishing more reliable data-driven surrogate models.

This study utilized a sequence of training sets with sample sizes of 50, 100, 150, 200, 300, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, and 2000 to fit the five surrogate models. The model trained with each sample size was evaluated on the test set with 400 samples. Fig. 16 shows the variation of the average prediction error of the model trained with different sample sizes on the entire test set. The subfigures in the first and the second row represent the results of the conventional and deep learning-based surrogate models, respectively. It could be found

that the prediction errors of the five models were large when the sample size was small. With the increase of training samples, the errors gradually decreased, and all tended to continue to decrease. The results showed that the sample size significantly affected the generalization performance of the model, and increasing the sample size could improve the generalization performance. For the U-bend problem in this study, at least 1000 samples were needed to train a model with good generalization accuracy. Additionally, the generalization accuracy of the CGAN model was less sensitive to the sample size compared with other models, and the prediction error quickly reached a low level, which indicated



**Fig. 14.** A quantitative comparison between the different surrogate models and CFD in the response of  $Q$  to different parameters.

that using the CGAN model for high-dimensional information modeling when the sample size was small may be a more reasonable choice.

More quantitatively, the RMSE of  $\Delta P$  and  $Q$  of different surrogate models trained under several typical sample sizes (i.e., 100, 400, 1000, 1600, and 2000) are given in Tables 4 and 5, respectively. For the  $\Delta P$ , the CGAN model and the ANN model had the smallest and largest sensitivity to sample size, respectively. As the sample size increased from 100 to 2000, the RMSE of the CGAN model and the ANN model decreased by 37.0% and 52.3%, respectively. For the  $Q$ , the ANN model and the CNN model had the smallest and largest sensitivity to sample size, respectively. As the sample size increased from 100 to 2000, and the RMSE of the ANN model and the CNN model decreased by 21.4% and 33.8%, respectively. In addition, the Kriging model and CGAN

model had obvious advantages when the sample size was small (i.e., 100 to 400). Also, the CGAN model was less sensitive to the sample size than the CNN model.

#### Comprehensive evaluation of the models

Besides the single-point accuracy, response accuracy and sensitivity to sample size discussed in the previous sections, the industry might also get interested in other characteristics of models, such as the GPU requirement, training cost, model size, etc. This section collected more characteristics in Table 6 to comprehensively evaluate the five presented models. It was found that the trainable parameters of the deep learning models were one to four orders of magnitude higher than that of the

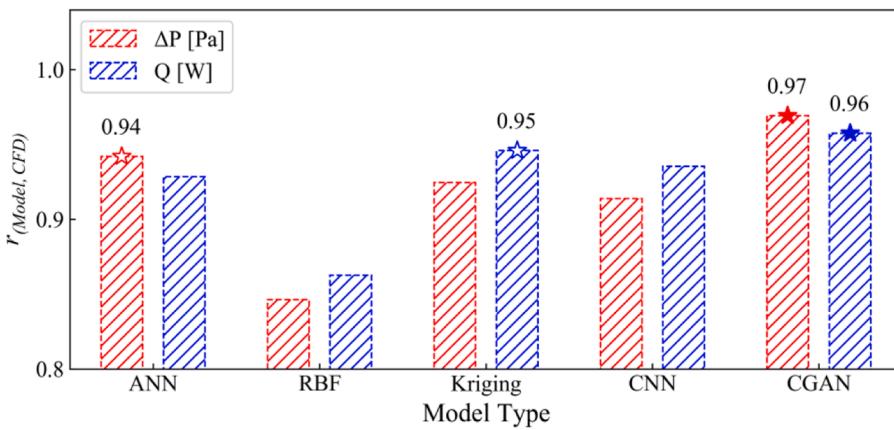


Fig. 15. A quantitative comparison of the response accuracy of different models.

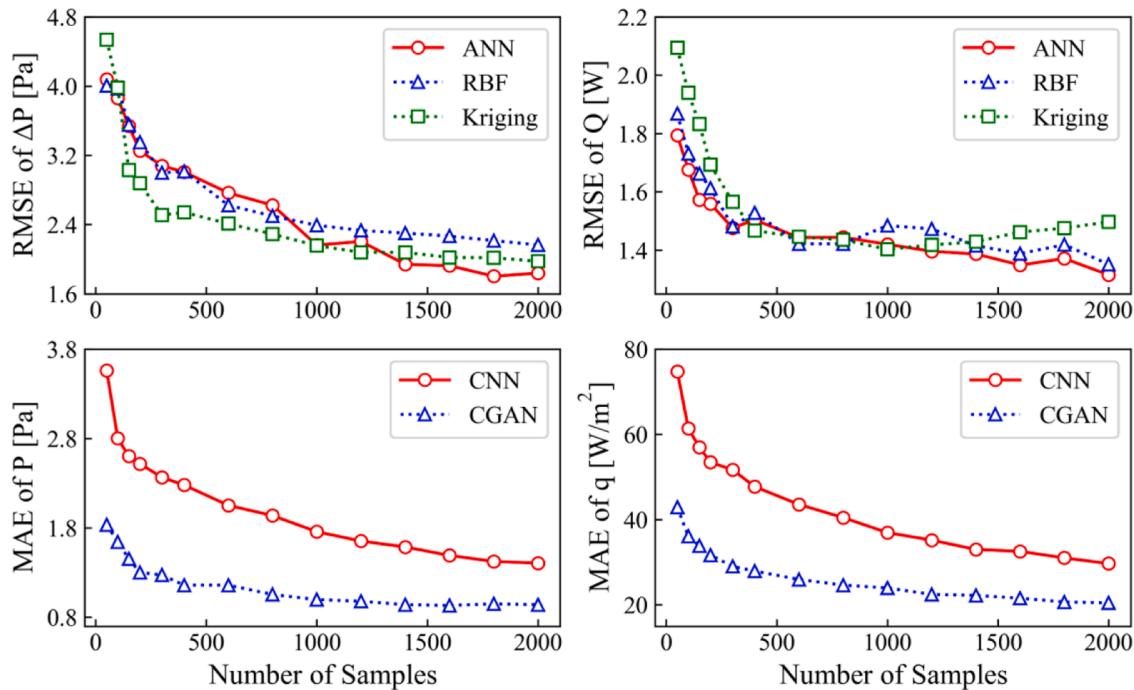


Fig. 16. Sensitivity of accuracy of different surrogate models to sample size.

Table 4

RMSE of  $\Delta P$  of different models trained under different sample sizes.

Sample size	ANN	RBF	Kriging	CNN	CGAN
100	3.86	3.97	3.98	4.37	3.81
400	3.01	3.01	2.54	3.26	2.73
1000	2.16	2.39	2.16	2.81	2.30
1600	1.93	2.27	2.02	2.73	2.74
2000	1.84	2.17	1.98	2.39	2.40
RMSE Reduction	52.3%	45.3%	50.3%	45.3%	37.0%

conventional models, which resulted in higher training time costs. Therefore, GPUs were usually required to accelerate the training of deep learning models. However, the prediction time costs of all surrogate models were on the order of ten milliseconds once the model was trained. The biggest advantage of the deep learning models was the ability to obtain 2-D images, which could provide designers with a reference to physical intuition. As compared with other models, the CGAN model predicted the objective variables more accurately, and

Table 5

RMSE of  $Q$  of different models trained under different sample sizes.

Sample size	ANN	RBF	Kriging	CNN	CGAN
100	1.68	1.73	1.94	2.25	1.89
400	1.50	1.53	1.47	1.95	1.63
1000	1.42	1.48	1.40	2.11	1.69
1600	1.35	1.39	1.46	1.61	1.58
2000	1.32	1.35	1.50	1.49	1.43
RMSE Reduction	21.4%	22.0%	22.7%	33.8%	24.3%

captured the responses best among the studied models. From the perspective of overall performance, the conventional models had higher prediction accuracy, while the deep learning models had higher response accuracy. In addition, the conventional and deep learning models were less sensitive to the sample size in predicting the  $\Delta P$  and  $Q$ , respectively. The ANN model and the CGAN model might be the best surrogate models for the problem focused in this study.

**Table 6**  
A comprehensive comparison of different surrogate models.

Evaluation indices	ANN	RBF	Kriging	CNN	CGAN
Trainable parameters	$3.7 \times 10^4$	$3.6 \times 10^3$	$\sim 2.0 \times 10^3$	$2.1 \times 10^5$	$5.7 \times 10^7$
Training time cost	1.3 h	0.8 h	18 h	2.7 h	16.7 h
Prediction time cost	$10^1$ ms	$10^1$ ms	$10^1$ ms	$10^1$ ms	$10^1$ ms
GPU requirements	×	×	×	✓	✓
Ability to provide 2-D information	×	×	×	✓	✓
Input requirements	Number	Number	Number	Number	Image
Prediction of $\Delta P$ (RMSE)	<b>1.84</b>	2.17	1.98	2.39	2.40
Prediction of $Q$ (RMSE)	<b>1.32</b>	1.35	1.50	1.49	1.43
Response of $\Delta P(r)$	0.94	0.85	0.92	0.91	<b>0.97</b>
Response of $Q(r)$	0.93	0.86	0.95	0.94	<b>0.96</b>
Sensitivity of $\Delta P$	52.3%	45.3%	50.3%	45.3%	<b>37.0%</b>
Sensitivity of $Q$	<b>21.4%</b>	22.0%	22.7%	33.8%	24.3%

## Conclusion

The present study conducted a detailed comparison between the conventional models and deep learning-based models in predicting the flow and heat transfer of U-bend channels. Based on the dataset obtained by numerical calculations, three conventional and two deep learning-based surrogate models were established. The prediction accuracy, response accuracy, sensitivity to sample size and other characteristics of interest of these five models were analyzed in detail. Results obtained from the modeling and analysis efforts indicated the following conclusions:

- 1 The two deep learning-based models both could well reconstruct the distributions of surface pressure and wall heat flux according to the given geometric information. The fast prediction of physical fields could realize real-time visualization of design results, which might assist designers.
- 2 The conventional models had higher single-point accuracy either in predicting pressure loss or heat transfer, but the deep learning-based models were also with acceptable prediction errors.
- 3 The deep learning-based models had higher response accuracy either in predicting pressure loss or heat transfer. Reconstruction of field information could better preserve the physics contained in the data.
- 4 The sample size had a significant influence on the generalization performance of all models. With the increase of training samples, the generalization errors gradually decreased.
- 5 Deep learning was a promising surrogate modeling approach due to its acceptable prediction error, good response accuracy and ability to reconstruct physical fields.

The conclusions are limited for this study and the selected U-bend definitions. The analysis could provide guidance for establishing more reliable and powerful data-driven surrogate models for the design of thermal equipment. According to the results obtained in this study, three suggestions could be made for surrogate modeling of thermo-fluid structure optimizations: 1) The Kriging model was more suitable for cases where the sample size was less than 300 and there was no need for physical field reconstruction; 2) The Artificial Neural Network was the best choice when the sample size was larger than 1000 and there was no need for image reconstruction; 3) If detailed physical fields needed to be visualized, the Conditional Generative Adversarial Network was an effective way even with less than 300 samples. Additionally, the fast physical field reconstruction function of deep learning methods could also be used for the artificial interaction design of thermo-fluid structures. Future efforts could be focused on how to introduce physics knowledge and design experience into data-driven modeling.

## Declaration of Competing Interests

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

## Acknowledgments

This work is supported by the National Science Foundation of China No. 51906139, the State Key Laboratory of Aerodynamics (SKLA-20190108), and the Shanghai Sailing Program (19YF1423200).

## References

- [1] Ding R, Yin W, Cheng G, Chen Y, Wang J, Wang R, et al. Boosting the optimization of membrane electrode assembly in proton exchange membrane fuel cells guided by explainable artificial intelligence. *Energy AI* 2021;5:100098.
- [2] Wang B, Zhang G, Wang H, Xuan J, Jiao K. Multi-physics-resolved digital twin of proton exchange membrane fuel cells with a data-driven surrogate model. *Energy AI* 2020;1:100004.
- [3] Haffejee RA, Laubscher R. Application of machine learning to develop a real-time air-cooled condenser monitoring platform using thermofluid simulation data. *Energy AI* 2021;3:100048.
- [4] Mondal S, Chattopadhyay A, Mukhopadhyay A, Ray A. Transfer learning of deep neural networks for predicting thermoacoustic instabilities in combustion systems. *Energy AI* 2021;5:100085.
- [5] Westermann P, Evins R. Using Bayesian deep learning approaches for uncertainty-aware building energy surrogate models. *Energy AI* 2021;3:100039.
- [6] Li, D., L. Qiu, K. Tao, and J. Zhu, Artificial intelligence aided design of film cooling scheme on turbine guide vane. *Propsuls Power Res*, 2020. 9(4): p. 344–354.
- [7] Maral H, Alpman E, Kavurmacioglu L, Camci C. A genetic algorithm based aerothermal optimization of tip carving for an axial turbine blade. *Int J Heat Mass Transf* 2019;143:118419.
- [8] Moon M-A, Kim K-Y. Analysis and optimization of fan-shaped pin-fin in a rectangular cooling channel. *Int J Heat Mass Transf* 2014;72:148–62.
- [9] Wang C, Zhang J, Zhou J. Optimization of a fan-shaped hole to improve film cooling performance by RBF neural network and genetic algorithm. *Aerospace Sci Technol* 2016;58:18–25.
- [10] Namgoong, H., C. Son, and P. Ireland, U-bend shaped turbine blade cooling passage optimization, in 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, 2010.
- [11] Verstraete T, Coletti F, Bulle J, Vanderwielen T, Arts T. Optimization of a U-bend for minimal pressure loss in internal cooling channels—part I: numerical method. *J Turbomach* 2013;135(5).
- [12] Coletti, F., T. Verstraete, T.e. Vanderwielen, J.r.m. Bulle, and T. Arts. Optimization of a U-bend for minimal pressure loss in internal cooling channels: part II—experimental validation. in ASME 2011 turbo expo: turbine technical conference and exposition. 2011.
- [13] Verstraete, T. and J. Li. Multi-objective optimization of a U-bend for minimal pressure loss and maximal heat transfer performance in internal cooling channels. in ASME turbo expo 2013: turbine technical conference and exposition. 2013.
- [14] Kim H-M, Moon M-A, Kim K-Y. Multi-objective optimization of a cooling channel with staggered elliptic dimples. *Energy* 2011;36(5):3419–28.
- [15] Kim K-Y, Shin D-Y. Optimization of a staggered dimpled surface in a cooling channel using Kriging model. *Int J Therm Sci* 2008;47(11):1464–72.
- [16] Kim S-M, Afzal A, Kim K-Y. Optimization of a staggered jet-convex dimple array cooling system. *Int J Therm Sci* 2016;99:161–9.
- [17] Seo J-W, Afzal A, Kim K-Y. Efficient multi-objective optimization of a boot-shaped rib in a cooling channel. *Int J Therm Sci* 2016;106:122–33.
- [18] Lee S, Hwang W, Yee K. Robust design optimization of a turbine blade film cooling hole affected by roughness and blockage. *Int J Therm Sci* 2018;133:216–29.
- [19] Seo HJ, Kang YJ, Lee HC, Kwak JS, Park JS, Lee KD. Optimization of the configuration of the laidback fan-shaped film cooling hole with a lateral expansion angle of 10 degrees. *Appl Therm Eng* 2019;153:379–89.
- [20] Wen J, Yang H, Tong X, Li K, Wang S, Li Y. Optimization investigation on configuration parameters of serrated fin in plate-fin heat exchanger using genetic algorithm. *Int J Therm Sci* 2016;101:116–25.
- [21] Yang L, Chen W, Chyu MK. A convolution modeling method for pore plugging impact on transpiration cooling configurations perforated by straight holes. *Int J Heat Mass Transf* 2018;126:1057–66.
- [22] Yang L, Min Z, Yue T, Rao Y, Chyu MK. High resolution cooling effectiveness reconstruction of transpiration cooling using convolution modeling method. *Int J Heat Mass Transf* 2019;133:1134–44.
- [23] Yang L, Dai W, Rao Y, Chyu MK. Optimization of the hole distribution of an effusively cooled surface facing non-uniform incoming temperature using deep learning approaches. *Int J Heat Mass Transf* 2019;145:118749.
- [24] Mirza, M. and S. Osindero Conditional generative adversarial nets. arXiv e-prints, 2014. arXiv:1411.1784.
- [25] Ma H, Zhang Y-x, Haidan OJ, Thurey N, Hu X-y. Supervised learning mixing characteristics of film cooling in a rocket combustor using convolutional neural networks. *Acta Astronaut* 2020;175:11–8.

- [26] Laubscher R, Rousseau P. Application of generative deep learning to predict temperature, flow and species distributions using simulation data of a methane combustor. *Int J Heat Mass Transf* 2020;163:120417.
- [27] Liu T, Li Y, Jing Q, Xie Y, Zhang D. Supervised learning method for the physical field reconstruction in a nanofluid heat transfer problem. *Int J Heat Mass Transf* 2021;165:120684.
- [28] Yang L, Wang Q, Rao Y. Searching for irregular pin-fin shapes for high temperature applications using deep learning methods. *Int J Therm Sci* 2021;161:106746.
- [29] Piegl L, Tiller W. Rational B-spline curves and surfaces. The NURBS Book. Berlin, Heidelberg: Springer Berlin Heidelberg; 1995. p. 117–39.
- [30] McKay MD, Beckman RJ, Conover WJ. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 1979;21(2):239–45.
- [31] Dosovitskiy, A., J.T. Springenberg, and T. Brox. Learning to generate chairs with convolutional neural networks. in 2015 IEEE conference on computer vision and pattern recognition (CVPR). 2015.
- [32] Isola, P., J. Zhu, T. Zhou, and A.A. Efros. Image-to-image translation with conditional adversarial networks. in 2017 IEEE conference on computer vision and pattern recognition (CVPR). 2017.