Energy and AI 20 (2025) 100499

Contents lists available at ScienceDirect

Energy and AI

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

Efficient estimation of convective cooling of photovoltaic arrays: A physics-informed machine learning approach

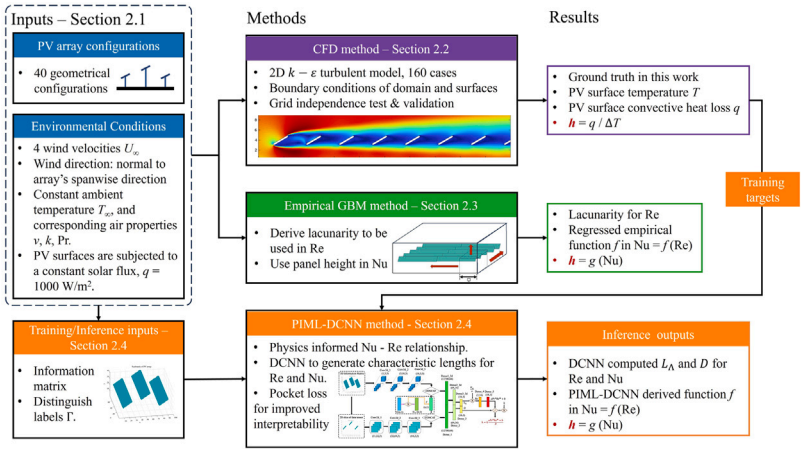
Dapeng Wang, Zhaojian Liang, Ziqi Zhang, Mengying Li *

Department of Mechanical Engineering & Research Institute for Smart Energy, The Hong Kong Polytechnic University, Hong Kong Special Administrative Region

HIGHLIGHTS

- A Physics-Informed Machine Learning model estimates heat dissipation in PV arrays.
- Data from 160 CFD simulations is used to train and validate the proposed model.
- A novel Pocket Loss function improves interpretability and model robustness.
- The model achieves 2.7% errors on testing datasets.
- The model outperforms empirical methods and is faster than CFD simulations.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:
Geometric configuration of PV array
Convective heat transfer
Deep convolution neural network
Physics informed machine learning
Pocket loss

ABSTRACT

Convective cooling by wind is crucial for large-scale photovoltaic (PV) systems, as power generation inversely correlates with panel temperature. Therefore, accurately determining the convective heat transfer coefficient for PV arrays with various geometric configurations is essential to optimize array design. Traditional methods to quantify the effects of configuration utilize either Computational Fluid Dynamics (CFD) simulations or empirical methods. These approaches often face challenges due to high computational demands or limited accuracy, particularly with complex array configurations. Machine learning approaches, especially hybrid learning models, have emerged as effective tools to address challenges in heat transfer design optimization. This study introduces a method that combines Physics-Informed Machine Learning with a Deep Convolutional Neural Network (PIML-DCNN) to predict convective heat transfer rates with high accuracy and computational efficiency. Additionally, an innovative loss function, termed the "Pocket Loss", is developed to enhance the interpretability and robustness of the PIML-DCNN model. The proposed model achieves relative estimation errors of 2.5% and 2.7% on the validation and test datasets, respectively, when benchmarked against comprehensive CFD simulations. These results highlight the potential of the proposed model to efficiently guide the configuration design of PV arrays, thereby enhancing power generation in real-world operations.

* Corresponding author.
E-mail address: mengying.li@polyu.edu.hk (M. Li).

https://doi.org/10.1016/j.egyai.2025.100499
Received 14 August 2024; Received in revised form 29 January 2025; Accepted 10 March 2025
Available online 22 March 2025

Nomenclature

Symbols

α	Coefficient of Pocket Loss
β_p	Temperature coefficient of photovoltaic panels, [%/K]
χ	Parameters need physical interpretation
ε	Relative error
Γ	Distinguish label in training, [m^2/s]
Λ	Lacunarity
ν	Kinematic viscosity of air, [N s/m^2]
Φ	Convolution Neural Network
Ψ	Geometric features
θ	Tilt angle of the panels, [rad], or Feed forward layers
ξ	Pocket Loss
C_{low}	Lower boundary in Pocket Loss
C_{up}	Upper boundary in Pocket Loss
D	Characteristic height of the array used to calculate Nu, [m]
d	Panel length, [m]
e	Maximum relative error
H	Height of the computation domain, [m]
h	Coefficient of convective heat transfer, [W/m^2 K]
k	Thermal conductivity of air, [W/m K]
L	Length of the computation domain, [m]
L_c	Characteristic length used to calculate Re, [m]
P	Generated power from PV, [W/m^2]
p_0	Ambient air pressure, [atm]
q	Heat flux, [W/m^2]
R^2	Coefficient of determination
r_i	Length of gliding box, [m]
r_{PO}	Power output ratio
S	Spacing between two rows of panels, [m]
$s(r)$	Ratio of volume occupied by gliding box
T	Temperature, [K]
t	Panel thickness, [m]
T_0	Initial temperature, [K]
T_∞	Ambient air temperature, [K]
U_∞	Wind velocity, [m/s]
W	Width of the computation domain, [m]
w	Weights of neural network
Nu	Nusselt number
Pr	Prandlt number
Re	Reynolds number

Subscripts

i	ith element or ith time
2D	Geometry in 2D scale
3D	Geometry in 3D scale
mod	PV panel module
STC	Standard test conditions
surf	Surface

Abbreviations

CFD	Computation Fluid Dynamics
DCNN	Deep Convolution Neural Network
GBM	Gliding Box Method
LES	Large Eddy Simulation
MAE	Mean Absolute Error
PIML	Physics-Informed Machine Learning
PL	Pocket Loss
PV	Photovoltaic
RMSE	Root Mean Square Error

1. Introduction

As one of the most prevalent devices for converting solar energy into electrical power, photovoltaic (PV) systems play a pivotal role in the transition towards sustainable energy sources. Effective thermal management is crucial for maintaining the performance of PV systems, as a 1 °C increase in panel temperature can decrease power output by 0.65% and conversion efficiency by 0.08% [1]. Consequently, the analysis and control of heat transfer in PV panels have become significant areas of research and development, particularly for large-scale installations such as PV arrays. The current methods for managing heat transfer are generally divided into passive cooling, active cooling, and hybrid cooling, most of which involve dissipating heat from the PV panel to an adjacent cooling medium [2]. For large-scale PV arrays, air-based convective cooling is particularly advantageous due to its cost-effectiveness and efficiency in removing substantial amounts of heat from panel surfaces. This method leverages the bulk movement of air, imposes minimal requirements on the thermal properties of the PV coating materials, and is highly suitable for desert regions where water resources are scarce [3].

For air-based convective cooling, the geometrical configurations of PV panels can significantly influence convective heat transfer performance, as demonstrated by Glick et al. [4]. In the context of single panels, Bilawane et al. [5] explored optimal inclination angles to enhance heat transfer efficiency. Additionally, Pawar and Sobhansarbandi [6] investigated the incorporation of phase change materials with heat pipe designs to optimize heat transfer. Pretorius and Nielsen [7] focused on design optimizations for both fixed-tilt and single-axis tracking PV systems, aiming to achieve optimal heat dissipation under both windy and calm conditions. For PV arrays, a staggered-height arrangement can improve airflow over subsequent panels, as shown in Fig. 1, and holds potential for optimizing heat transfer design in PV systems [8].

Accurate quantification of the impacts of array configurations on the convective heat transfer coefficient under various environmental conditions necessitates extensive Computational Fluid Dynamics (CFD) simulations [9], complemented by wind tunnel and field experiments [4,10]. However, these methods are resource-intensive. For instance, Large Eddy Simulations (LES) require approximately 43,000 core hours on a high-performance computing cluster to compute six cases of a LES transient study using the Uintah MPMICE software [11]. The substantial computational cost and hardware demands associated with CFD pose significant challenges for practical engineering applications. To facilitate the quantification of heat transfer performance in PV arrays and enhance their convective heat transfer, researchers have developed empirical correlations incorporating geometric simplifications. Glick et al. [10] simplified PV array configurations into flat plate arrays to establish a relationship between the Nusselt number (Nu) and Reynolds number (Re) based on wind tunnel experiments. Jaffer [12] applied boundary-layer theory to construct an empirical relationship among dimensionless numbers, quantifying natural convection heat transfer for isothermal plates. To address complex configurations in PV arrays, some studies have introduced and developed the concept

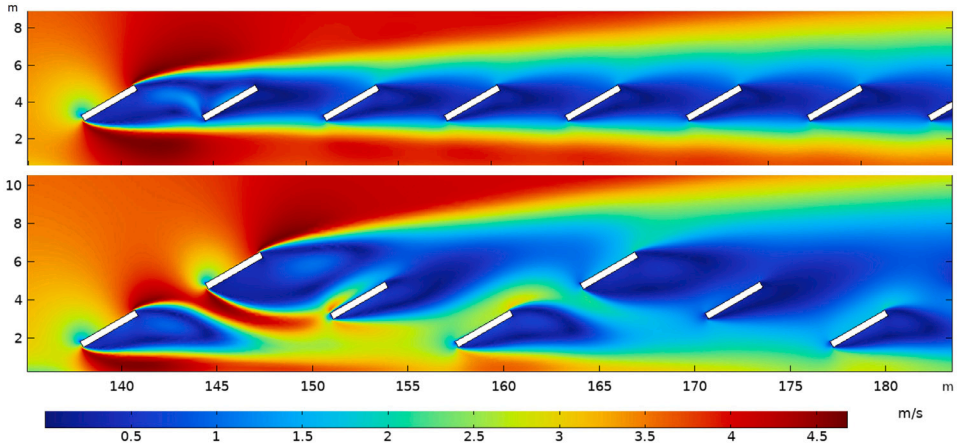


Fig. 1. Wind velocity field around PV arrays with a uniform-height configuration (upper plot) and a staggered-height configuration (lower plot). The results were generated using COMSOL.

of spatial distribution lacunarity, which effectively characterizes the configurations of PV panels within an array [13]. Nonetheless, challenges in determining characteristic length scales and limitations due to small sample sizes compromise the accuracy and generalizability of these empirical correlations.

In summary, current methodologies for quantifying convective heat dissipation in PV arrays present a significant trade-off: empirical approaches are computationally efficient but often lack precision, whereas Computational Fluid Dynamics (CFD) methods, while accurate, are computationally intensive. Thus, this work addresses the urgent need for a method that can estimate convective heat transfer in PV arrays with various configurations, achieving both high accuracy and computational efficiency. Consequently, we propose a Machine Learning (ML)-based approach to estimate the convective heat transfer of various PV array configurations. This method aims to combine the computational efficiency of empirical models with the accuracy of detailed simulations, providing a robust solution for optimizing heat dissipation and improving the overall efficiency of PV systems.

For heat transfer and energy-related applications, machine learning methods demonstrate substantial advantages over traditional physical approaches such as simulation and experimentation. These advantages are evident in terms of accuracy and computational efficiency [14]. The integration of physical laws into machine learning — termed physics-informed machine learning (PIML) — significantly enhances the interpretability of machine learning applications in complex physical systems [15,16]. For instance, Efatinasab et al. [17] developed both machine learning and deep learning models to quantify the performance of heat exchangers equipped with micro-finned tubes for design optimization purposes. Hughes et al. [18] developed a machine learning model with physically informed features to predict real-time temperatures of electronic machines. Image analysis techniques, such as DCNN and other methods, have become increasingly popular in energy and heat transfer research. These approaches have shown considerable potential for quick solution estimation in physical problem-solving domains, including the modeling of hydrological systems and turbulent heat transfer [19]. Yang et al. [20] employed a graphical neural network in conjunction with compact physical equations to model the thermal dynamics of buildings. Moreover, hybrid machine learning approaches that combine PIML with deep learning hold significant promise for providing precise and rapid estimations of convective heat transfer coefficients for design optimization [14]. Kalpana et al. [21] developed an innovative hybrid model combining a Physics-Informed Neural Network with a Convolutional Neural Network to predict the heat transfer coefficient in complex configurations involving gas-solid fixed beds and coal combustion and gasification, achieving a coefficient of determination (R^2) of 0.98.

In the PV sector, these techniques are applied in areas such as solar energy forecasting [22–24], panel defect detection, and addressing the uncertain characteristics of energy sources to ensure stability in energy generation [16]. These successful applications provide a solid foundation of inspiration for our work. However, there is a notable gap in the literature regarding the specific application of hybrid ML models to optimize PV array designs. Additionally, there is a lack of research focused on enhancing the interpretability of PV configuration optimization models during the training process, rather than exclusively in the final results. This study aims to fill these gaps by exploring the use of PIML combined with DCNN to enhance the heat transfer performance within PV arrays, thus optimizing their design. Our model employs a DCNN to process physical information from PV arrays, making it robust enough to handle different geometrical configurations and environmental conditions. Furthermore, we have improved the interpretability of the model by proposing a new loss function called "Pocket Loss" for PIML, which provides physical insights into the model's estimation. In the following sections, Section 2 presents the methodology for benchmark models and the proposed PIML-DCNN model. Section 3 presents the comparative results of the models considered. The concluding remarks are summarized in Section 4.

2. Methodology

In this work, three methods are used and compared to estimate heat transfer in PV arrays: an empirical method [13], a validated CFD model, and the proposed PIML-DCNN. The results of the CFD model serve as the ground truth for training and testing the PIML-DCNN model, as well as the benchmark for evaluating the accuracy of the empirical method. For all three methods, the inputs are the configurations of the PV array and the environmental conditions, including panel heights, row spacings, wind velocities, air properties, and incident solar flux. The outputs are the averaged convective heat transfer coefficients h of all panels. The methodology flowchart is presented in Fig. 2.

2.1. Problem description

Under the conditions of a constant ambient temperature T_∞, a constant solar flux incident on PV panels, and a given wind speed U_∞, the configuration of PV array – especially the row spacing S and the height H of each row – play a pivotal role in determining the turbulent flow around the panels and the convective heat transfer rate [13]. In

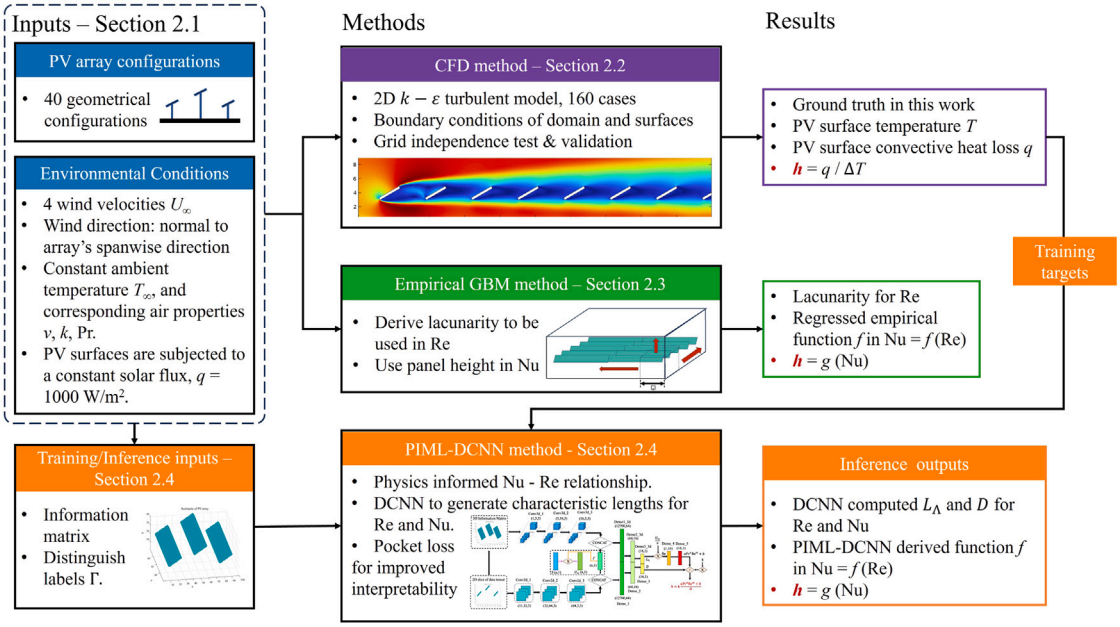


Fig. 2. Methodology flowchart of this work. The inputs are PV array configurations and environmental conditions. The outputs are the convective heat transfer coefficients. The methods compared include a CFD method, an empirical method, and the proposed PIML-DCNN method.

this study, the arrays are categorized into two different height configurations: the same-height array, where all rows have the same height, and the staggered-height array, which consists of rows with varying heights, as shown in Fig. 3. To analyze the convective heat transfer rate under various conditions, the height arrangement, row spacing S, and wind speed U_∞ are selected as manipulated variables. Variations in the spanwise direction are assumed to be negligible [13]. A total of 160 cases ($8 \times 5 \times 4 = 160$) are analyzed based on the configurations outlined in Fig. 3, with each case being a unique combination of height arrangement (8 cases), S (5 cases), and U_∞ (4 cases) . Other configuration parameters are determined according to the dimensions reported in [13]. A constant solar irradiance of 562.5 W/m² is applied to the upper surface of all PV panels [10], with 112.5 W/m² being converted into electricity and the remaining 450 W/m² being dissipated through convection. The 450 W/m² of absorbed solar energy translates to an internal heat source with an intensity of 1286 W/m³, considering the dimensions of PV panels.

2.2. The CFD method

A CFD model has been developed in COMSOL Multiphysics to simulate both velocity and temperature fields by solving the momentum and energy equations. To primarily address the effects of turbulence on heat transfer, rather than focusing on the detailed dynamics, the $k-\varepsilon$ turbulence model is employed. The boundary conditions for these CFD simulations are detailed in Table 1. The fluid is assumed to be incompressible, with its pressure set to $p_\infty = 1$ atm. The open boundary condition indicates that the normal stress is zero. The dimensions of the computational domain are $L = 300$ m and $H = 53$ m, and the ambient and initial temperatures of the PV module are set to 300.15 K, as described in Stanislawski et al. [11]. The average convective heat transfer coefficient, h, for the array is determined as:

$$h = \frac{\bar{q}_s}{\bar{T}_{\text{surf}} - T_\infty} \tag{1}$$

where $\bar{q}_s$ and $\bar{T}_{\text{surf}}$ represent the average convective heat flux and surface temperature across all panel surfaces, respectively.

The validation of the CFD results is detailed in Appendix. Following this validation, the CFD model was utilized to generate a dataset consisting of 160 distinct cases, each varying in panel heights, row spacings, and wind velocities. From this dataset, 100 cases were selected to train the proposed PIML-DCNN model, while another 40 cases were reserved for model validation. The remaining 20 cases, which represent the Low-Medium-High (LMH) configuration – a particularly complex configuration not included in the training set – were used for further test of the PIML-DCNN model.

2.3. The empirical method

To estimate the convective heat transfer of PV arrays with inclined PV panels, the following empirical correlation is proposed by Smith et al. [13],

$$\log_{10}(\text{Nu}) = a\text{Re}^m\text{Pr}^n + b \tag{2}$$

where a, b, m, and n are empirically fitted coefficients. Pr is the Prandtl number, and Nu is the Nusselt number, representing the ratio between heat convection and conduction,

$$\text{Nu} = \frac{hD}{k} \tag{3}$$

where, h is the convective heat transfer coefficient, and k is the thermal conductivity of air. D is the characteristic height of the array, which is represented by the canopy height of the panel at the 8th row, where the flow boundary is fully developed [13]. Re is the Reynolds number, representing the ratio between inertial and viscous forces,

$$\text{Re} = \frac{U_\infty L_c}{\upsilon} \tag{4}$$

where U_∞ represents the wind velocity, L_c is the characteristic length, and υ is the kinematic viscosity of air.

To ensure the effectiveness of the above empirical equation, it is essential to derive a characteristic length scale, L_c, to represent the configurations of PV arrays, where PV panels are not homogeneously distributed. Therefore, Smith et al. [13] proposed a method to derive L_c based on the concept of lacunarity. Lacunarity is quantified by the Gliding Box Method (GBM), where multiple boxes with different lengths are introduced to move through the entire PV array space, as shown in Fig. 4 [13]. GBM updates the PV configuration's occupation features within the box in each step and generates an overall lacunarity

Table 1
Boundary conditions of the CFD model. The computation domain is $L = 300\,\text{m}$ and $H = 53\,\text{m}$.

	$x = 0$ (inlet)	$x/L = 1$ (outlet)	$z = 0$ (bottom)	$z/H = 1$ (top)	PV body
Momentum	$U{=}U_\infty$	Open Boundary	$U = 0\,\text{m/s}$	Open Boundary	–
Thermal	$T_\infty = 300.15\,\text{K}$	$T_0 = 300.15\,\text{K}$	Zero flux	$T_0 = 300.15\,\text{K}$	$q = 1286\,\text{W/m}^3$

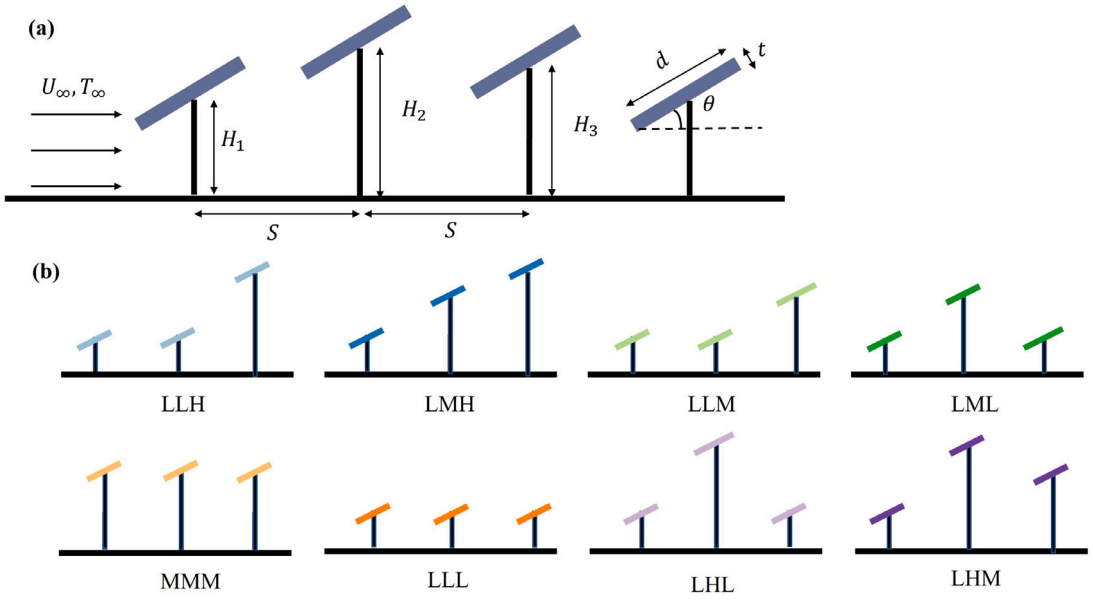


Fig. 3. (a) Schematic of the PV array configurations with the inflow air velocity U_∞, ambient temperature T_∞, and row spacing S. Five different row spacings are considered: 5.81 m, 6.54 m, 7.26 m, 7.99 m, and 8.72 m. The panel dimensions include a length $d = 3.3$ m, a thickness $t = 0.35$ m, and a tilt angle $\theta = 30°$. The variables H_1, H_2, and H_3 represent different heights from the ground to the center of the panel. (b) Investigated height arrangement: 'L' denotes 'Low height' with $H_1 = 1.52$ m, 'M' denotes 'Medium height' with $H_2 = 3.00$ m, and 'H' denotes 'High height' with $H_3 = 4.56$ m.

dimensionless number, Λ, from all the updates. Then, the characteristic length L_c can be represented as,

$$L_c = \frac{\sum(\Lambda \times R)}{N_R}, \quad R = [r_1, r_2, \ldots, r_N] \tag{5}$$

where R represents the set of box sizes used to calculate lacunarity, Λ is the corresponding lacunarity of different boxes, r_N is the resolution of the gliding box that should be no larger than half of the smallest length of a PV panel, and N_R is the number of utilized box with different sizes [13]. Before calculating L_c, an independence test is performed on the value of r_N to ensure that changes in its value do not influence the calculation of L_c, which is detailed in Appendix. In our experiments, each configurations in Fig. 3 undergoes an independence test prior to the development of the empirical method.

Before calculating L_c, an independence test is conducted on the value of r_N to ensure that changes in this parameter do not influence the calculation of L_c. For our experiments, each array configuration undergoes an independence test prior to the development of the empirical method. More details about the independence test is provided in Appendix.

For the 160 cases we considered, the characteristic length and height obtained using the GBM are presented in Fig. 5. The results show that the characteristic length ranges from 4 to 10 m, while the characteristic height ranges from 3 to 7 m, respectively.

In addition to the correlation presented in Eq. (2), another widely used empirical formula for heat convection over an inclined surface is given by [25–27]:

$$\text{Nu} = a\text{Re}^m\text{Pr}^n + b \tag{6}$$

This model will also serve as a benchmark for evaluating the performance of the proposed PIML-DCNN model.

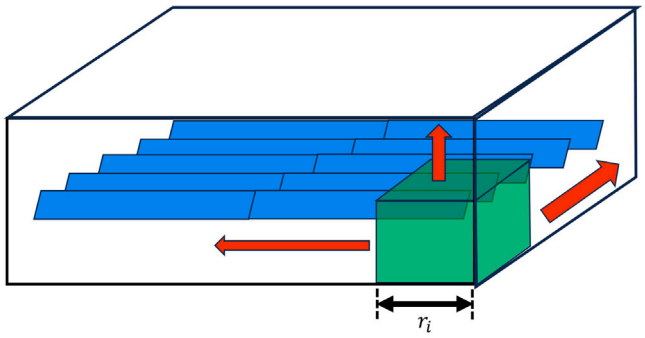


Fig. 4. The schematic illustration of the grid-based method (GBM) for calculating lacunarity, where r_i represents the length of the box used to quantify Λ at the current step.

2.4. The PIML-DCNN method

Although the CFD model can achieve satisfactory levels of accuracy, it requires substantial computational resources. In contrast, the empirical model offers efficient estimations but at the cost of reduced accuracy. To enhance the accuracy of the empirical model significantly, it is vital to capture complex geometric configurations effectively. The DCNN presents a robust alternative to GBM for this task, as both methods utilize iterative updates within their kernels to process geometric information. In this context, we introduce the PIML-DCNN as a solution that achieves a balance between high accuracy and computational efficiency.

In this methodology, a DCNN is employed to extract characteristic lengths L_c and D from array configurations with both same-height

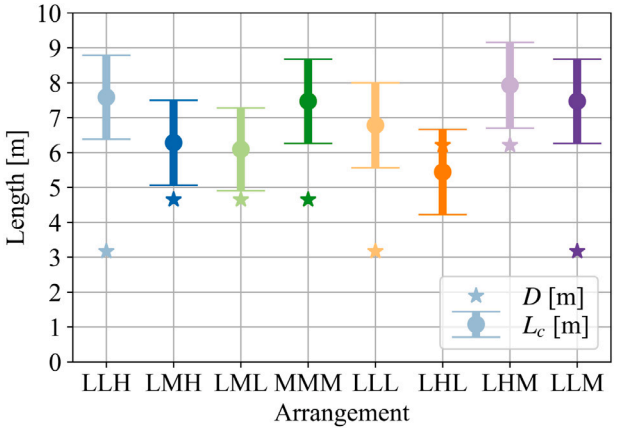


Fig. 5. Characteristic length, L_c, and characteristic height, D, derived from the GBM across all investigated geometric configurations. This data will also be utilized in developing the Pocket Loss function for the PIML-DCNN model.

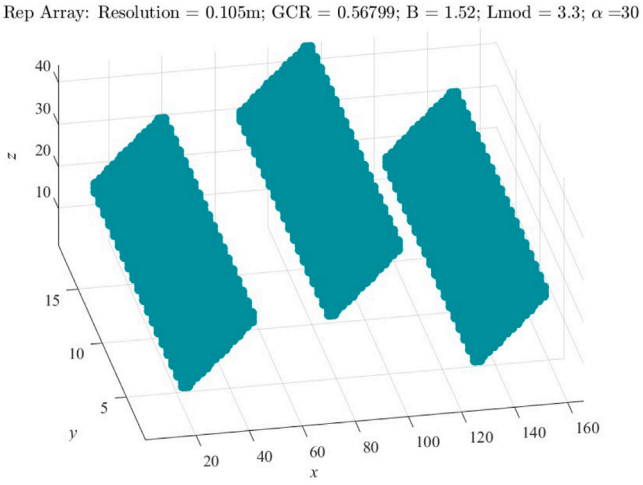


Fig. 6. Illustration of the 3D information matrix representing three rows of PV panels. The projection of this matrix onto the x–y plane results in the corresponding 2D slice matrix.

and staggered-height arrangements. Specifically, L_c is determined using a three-dimensional (3D) DCNN, while D is derived from a two-dimensional (2D) DCNN. Subsequently, L_c and D are used to calculate the Nusselt number and Reynolds number. Additionally, a multiple dense layer structure is designed to capture the empirical correlation between Nu and Re, as expressed in Eq. (6). This correlation, informed by physical principles, is instrumental in estimating the convective heat transfer coefficient h.

2.4.1. Model inputs and outputs

The input data for the PIML-DCNN model include both environmental conditions (i.e. wind speed) and configurations of PV arrays (i.e. height arrangement and spacing). The model outputs the averaged convective heat transfer coefficient h across all panels. The primary learning objective is to minimize the error between the estimated h and the h calculated by the CFD model.

The array configurations are imported as an information matrix, constructed at a resolution that is half of the minimum dimension among all input features, in accordance with the guidelines proposed by [13]. This matrix encapsulates essential characteristics of the PV arrays, such as the number of rows, panel dimensions, and the spacing between rows. Within this matrix, geometric shapes are represented by discrete points, with a '1' indicating the presence of a panel and a '0' denoting its absence. These points are plotted in a three-dimensional space to form the information matrix, which graphically represents the layout of the PV panels, as depicted in Fig. 6. It is important to note that for our PIML-DCNN model, configurations of 10 rows are employed, as opposed to the three-row configuration illustrated in Fig. 6. The span length of the panel in the information matrix is configured to be 2.0 m, aligning with the specifications referenced in [11].

Although the information matrix effectively represents the array configuration with high fidelity, the features of wind turbulence also depend on wind velocity. To address this variable in our machine learning model, we adopt the feature-crossing strategy, which involves the combination of features through multiplication, as described by Luo et al. [28]. Consequently, we define a new parameter, Γ, as the product of wind velocity U_∞ and row spacing S:

$$\Gamma = S \times U_\infty \tag{7}$$

This parameter Γ is introduced to account for the turbulent field features and to enhance the influence of row spacing and wind speed.

2.4.2. PIML-DCNN model structure

The inputs to the PIML-DCNN model include a 3D information matrix, a 2D slice matrix derived as the projection of the information matrix on the wind field plane, the parameter Γ, the ratio U_∞/υ, and the thermal conductivity of air k. As illustrated in Fig. 7, the 3D information matrix is processed through a 3D DCNN structure and subsequently through feed-forward layers, denoted as θ, to calculate the characteristic length L_c,

$$L_c = \theta_{\text{Dense1\_3d}-\text{Dense3\_3d}}(\boldsymbol{\Phi}_{3D}(\boldsymbol{\Psi}_{3D})) \tag{8}$$

Similarly, the 2D slice matrix undergoes processing to produce the characteristic height D:

$$D = \theta_{\text{Dense1}-\text{Dense3}}(\boldsymbol{\Phi}_{2D}(\boldsymbol{\Psi}_{2D})) \tag{9}$$

During the feed-forward process in both the 3D and 2D DCNN architectures, after convolution, the parameter Γ is integrated with the outputs of the 3D and 2D convolution layers. This integrated data forms the input for subsequent layers, which compute L_c and D. L_c is then combined with U_∞/υ to calculate the Reynolds number, Re. This value is processed through Dense layers 4 and 5 to approximate the empirical correlation outlined in Eq. (6) [29]:

$$\text{Nu} = a\text{Re}^m\text{Pr}^n + b = \left(\prod_{i=1}^{2} w_{3-i}^{\text{T}}\right)\text{Re} + b \tag{10}$$

A detailed proof of this approach is presented in Appendix. Ultimately, h is calculated using Nu, D and k.

Inspired by the U-Net architecture, which has proven effective in processing low-resolution images [30], our model adopts a similar strategy for managing information matrix data, characterized as low-resolution imagery. Each convolutional layer is followed by a max-pooling layer to efficiently down-sample the data.

2.4.3. Pocket loss function

Although the current PIML-DCNN model can accurately approximate h across the considered 160 cases, the resulting values for L_c range from −6.08 m to −5.27 m and for D, from 852 m to 1019 m. These ranges significantly exceed those with physical relevance, as presented in Section 2.3. The non-physical nature of L_c and D suggests the limited interpretability of the model, particularly with respect to the Nusselt number. To enhance the interpretability of PIML, it may be beneficial to customize loss functions by incorporating physical laws directly into the loss function [31], and to explore other innovative

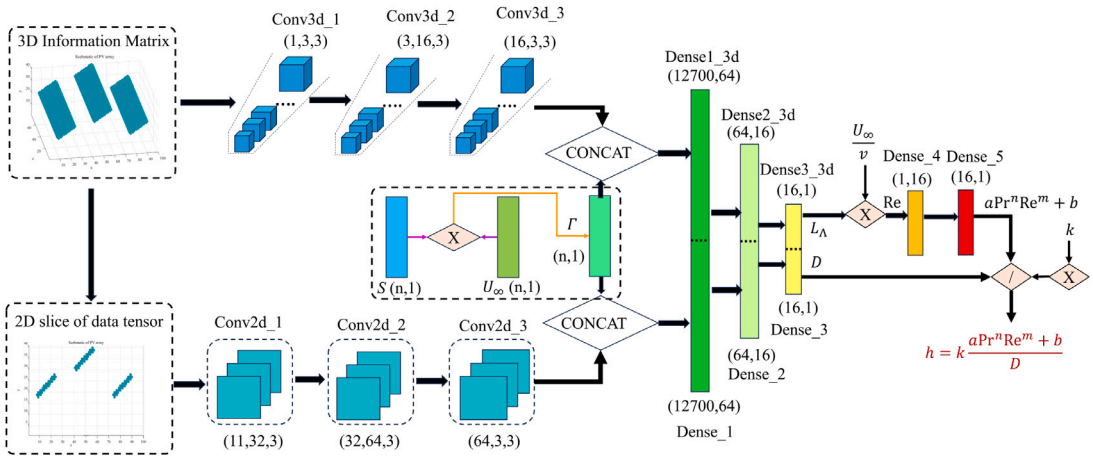


Fig. 7. Structure of the proposed PIML-DCNN model. Each tuple represents the size of the respective layer, with the last element indicating the kernel size of the DCNN layers.

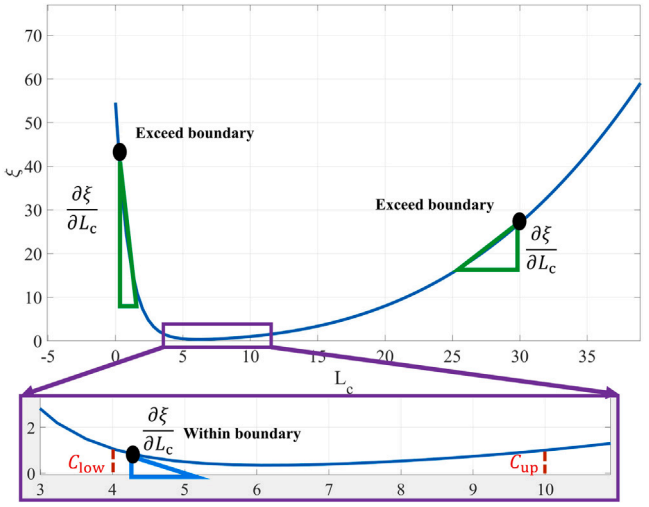


Fig. 8. Illustration of Pocket Loss for L_c Using Gradient Representation. This plot depicts the loss values as dots, with the corresponding gradients represented by triangles: green triangles indicate values that exceed the boundary, while blue triangles show values within the boundary. Dashed lines delineate the lower and upper boundary values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

approaches like those proposed in Rao et al. [32]. Consequently, we introduce a novel loss function, termed "Pocket Loss", to enhance the interpretability of the proposed model:

$$\xi_i = \frac{\bar{\chi}_i^3}{C_{\text{up}}{}^3} + \frac{1}{e^{\bar{\chi}_i - C_{\text{low}}}} \tag{11}$$

where ξ_i represents the Pocket Loss for the physical parameter requiring interpretability; $\bar{\chi}_i$ denotes the average value of χ_i. The constants C_{up} and C_{low} are the upper and lower boundary values, respectively, based on the physical knowledge of the domain for χ_i. We define Pocket Loss in this way to maintain the model's overall convexity during training – particularly important for generating negative outputs – and to robustly encourage exploration around the defined boundary values. This is achieved as the gradients near the boundaries are significantly lower than those farther away, guiding the model to stay within realistic limits.

Taking L_c as an example, its range is approximately between 4 m and 10 m, as depicted in Fig. 5. We set the constraints with C_{up} = 10 m and C_{low} = 4 m to ensure that L_c remains within this specified range. As illustrated in Fig. 8, during training, values that exceed the boundaries are subjected to a higher magnitude and steeper loss gradients compared to values that remain within the boundaries, where the loss remains below 1.0. Consequently, as the deep learning model strives to minimize the loss value and its gradient, the upper and lower boundaries act as a 'pocket', containing the variations in L_c and its gradient. This pocket guides the model towards the defined range, resulting in relatively low values of ξ and $\partial\xi/\partial L_c$. Additionally, Pocket Loss remains differentiable within the boundary region, which circumvents issues related to non-convexity and local minima, thus ensuring a smooth and robust training process.

The boundary value settings for D are also based on the information presented in Fig. 5, with C_{up} set to 7 m and C_{low} set to 3 m. When interpretability is required for multiple physical parameters, Pocket Loss is redefined as,

$$\xi = \sqrt{\sum_{i=1}^{n}\left(\frac{\bar{\chi}_i^3}{C_{\text{up},i}{}^3} + \frac{1}{e^{\bar{\chi}_i - C_{\text{low},i}}}\right)^2} \tag{12}$$

This equation directs the adaptation of all parameters during training by minimizing the projected distance within an n-dimensional geometric space, where n represents the number of physical parameters requiring interpretability. The refined loss function for training the model is subsequently defined as follows:

$$\text{Loss} = \text{RMSE} + \alpha\xi \tag{13}$$

where RMSE represents the Root Mean Square Error, quantifying the discrepancy between the model's predicted h and the actual h (from CFD simulations). The coefficient α is introduced to balance the gradient between ξ (the interpretability term) and RMSE during training. For our experiments, we set the value of α to be 0.3 based on trial and error experiments. Details of each model component for models with Pocket Loss and without Pocket Loss are summarized in Table 2.

2.4.4. Training process of PIML-DCNN model

In this work, 160 cases were produced using the validated CFD model. Out of these, 100 cases were used to train the PIML-DCNN model, while 40 cases were designated for validation and 20 cases comprised the testing dataset. During the training process, 20 cases are combined into a single batch and inputted into the model for simultaneous training. To optimize the smoothness and stability of the training process, a total of 300 epochs were implemented. During the initial 50 epochs, a learning rate of 10^{-3} was applied, utilizing the Adam optimizer and early dropout techniques to quickly guide the model towards the pre-established boundaries. Subsequently, the learning rate was reduced to 10^{-4} for the next 110 epochs. To counteract underfitting, especially when training with limited data, we incorporated an

Table 2
Details of each model component, including layer name, output shape, arguments settings, and the number of parameters to be trained. Each MaxPool is followed by a ReLU function. As Pocket Loss performs similar role as ReLU and PReLU that eliminate value lower than 0 during training process, so for model without PL Dense3_3d /Dense_3 will follow a PReLU function.

Layer name	Output shape	Hyperparameter	Parameter number
Conv2d_1	[−1, 32, 498, 35]	stride = 1, kernel = (3,3)	896
MaxPool_2d		kernel = (2,2)	0
Conv2d_2	[−1, 64, 249, 17]	stride = 1, kernel = (3,3)	18,496
MaxPool_2d		kernel = (1,1)	0
Conv2d_3	[−1, 3, 249, 17]	stride = 1, kernel = (3,3)	1731
MaxPool_2d		kernel = (1,1)	0
Conv3d_1	[−1, 3, 3, 498, 35]	stride = 1, kernel = (3,3)	84
MaxPool_3d		kernel = (2,2)	0
Conv3d_2	[−1, 16, 1, 249, 17]	stride = 1, kernel = (3,3)	1312
MaxPool_3d		kernel = (1,1)	0
Conv3d_3	[−1, 3, 1, 249, 17]	stride = 1, kernel = (3,3)	1299
MaxPool_3d		kernel = (1,1)	0
Flatten_1	[−1, 12699]		0
Flatten_2	[−1, 12699]		0
Dense1_3d/ Dense_1	[−1, 64]		812,864
Dense2_3d /Dense_2	[−1, 16]		1040
Dense3_3d /Dense_3			
(With PReLU if no PL)	[−1, 1]		17
Dense_4	[−1, 16]		32
Dense_5	[−1, 1]		17
Total parameters number:			1,651,709

'Early Dropout and Late Drop' technique [33]. The dropout layer is active only during the initial and final 20% of the training epochs and is subsequently deactivated to address both overfitting and underfitting issues. These adjustments were aimed at fine-tuning the model to maintain optimal accuracy on both the test and validation datasets, ensuring performance within or close to the established boundaries. Ultimately, the performance of the trained model was evaluated on the validation and testing dataset to assess the applicability of the proposed PIML-DCNN model. For comparison purposes, the model without PL will be trained using the same processes, except for the 'Early Dropout and Late Drop' technique.

3. Results and discussion

3.1. Performance of the empirical method

To assess the performance of both the empirical method and the PIML-DCNN method when taking the CFD results as ground truth, the coefficient of determination (R^2) and the maximum relative error (ϵ) are utilized as metrics, which is defined as:

$$\epsilon = \left(\frac{\mathrm{MAE}(\hat{y}, y)}{\min(\bar{y}, \bar{\hat{y}})} \right) \times 100\% \tag{14}$$

In this context, $\hat{y}$ denotes the predicted value of h, y represents the value obtained from the CFD simulation (ground truth), $\bar{y}$ and $\bar{\hat{y}}$ are the average values of y and $\hat{y}$, respectively, and MAE is the Mean Absolute Error between the predictions and the target values.

As shown in Table 3, upon applying the empirical model, as detailed in Eqs. (2) and (6), to our CFD dataset, the highest model accuracy was achieved with the parameters $a = 0.09$, $b = 1.91$, $m = 0.2$, and $n = 0.0833$ for Eq. (2), and $a = 0.6093$, $b = 1.0597$, $m = 0.6336$, and $n = 1.3322$ for Eq. (6). Despite this optimization, the overall accuracy remains unsatisfactory, with an R^2 of 0.61 and an ϵ of 17%. This level of accuracy indicates that the characteristic lengths employed in the empirical model may not sufficiently capture the complexity of PV array configurations.

3.2. Performance of the PIML-DCNN model

As presented in Fig. 9 and Table 3, the PIML-DCNN model utilizing Pocket Loss demonstrates exceptional performance. In both the validation and test datasets, it achieves relative errors of 2.5% and 2.7%, respectively, with R^2 values of 0.99. This indicates that the proposed PIML-DCNN model is significantly more accurate compared to the empirical model.

Furthermore, Table 3 demonstrates that the introduction of the Pocket Loss function enhances both the interpretability and the physical plausibility of the PIML-DCNN model. By incorporating Pocket Loss, the model ensures that the characteristic length L_c and characteristic height D are constrained within a physically plausible ranges of 4.47 to 5.92 m and 4.99 to 5.80 m, while maintaining the characteristic length L_c within a range of 3.35 to 4.24 m. Additionally, the Reynolds numbers predicted by the model incorporating Pocket Loss fall within a range of 4.7×10^5 to 1.4×10^6, closely aligning with the valid range established by Smith et al. [13].

Moreover, the proposed PIML-DCNN model requires significantly less computational time compared to CFD approaches. For the CFD model used in this study, a single case employing the $k-\epsilon$ turbulence model requires at least 3 min to produce reasonable results on a high-performance AMD Ryzen Threadripper 3970X 32-Core Processor. In contrast, our trained PIML-DCNN model is capable of estimating the heat transfer performance of various configurations almost instantaneously.

Fig. 10 presents a visual comparison between the estimations of the convective heat transfer coefficient h from both the CFD simulations and the PIML-DCNN model. The heat maps, indicating the average h across the array from CFD results and model predictions, show a high level of concordance, validating the accuracy of the PIML-DCNN model. Additionally, Fig. 10 reveals that staggered height configurations encounter more significant heat transfer challenges than uniform height scenarios, with the optimal arrangement being when all panels are at the same height of 3.3 m. However, the subpar performance of staggered-height configurations contrasts with their observed beneficial impact on regional flow optimization, as seen in Fig. 1. To explain this inconsistency, a more in-depth analysis of the CFD results is undertaken in the following section.

3.3. Comparison between panel configurations

The efficiency and power output of PV panels are markedly affected by their temperature. To evaluate this influence, we present the power output ratio of panels under various configurations compared to their performance under standard conditions in Fig. 11. The power output ratio is calculated as:

$$r_{\mathrm{PO}} = \frac{P}{P_{\mathrm{STC}}} = 1 + \beta_p (T_{\mathrm{mod}} - T_{\mathrm{STC}}) \tag{15}$$

Table 3
Comparison of prediction accuracy for the convective heat transfer coefficient h and interpretability based on the value ranges of L_c and D across various methods.

	Correlation Eq. (2)	Correlation Eq. (6)	PIML-DCNN (no PL)	PIML-DCNN (with PL)
ϵ (%) (validation)	27%	37%	3.4%	2.5%
ϵ (%) (test)	19%	39%	6.2%	2.7%
R^2 (validation)	0.28	−0.1	0.99	0.99
R^2 (test)	0.49	−0.48	0.94	0.99
L_c	[4.23 m, 9.16 m]	[4.23 m, 9.16 m]	[−6.08 m, −5.27 m]	[4.47 m, 5.92 m]
D	[3.17 m, 6.21 m]	[3.17 m, 6.21 m]	[850 m, 1019 m]	[4.99 m, 5.80 m]

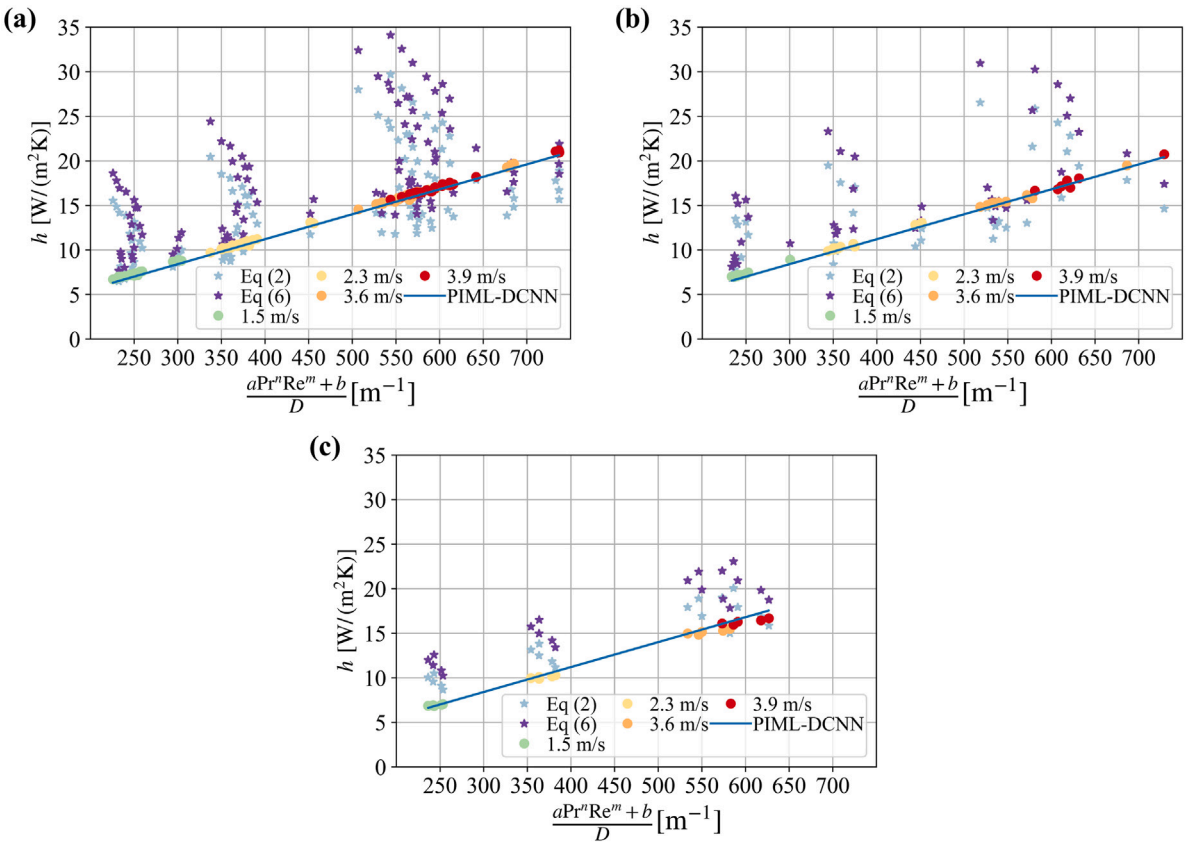


Fig. 9. Performance of the proposed PIML-DCNN model with Pocket Loss function on (a) training data, (b) validation data, and (c) testing data. Colored dots represent CFD results under various wind speeds, stars indicate results from the empirical method, and the line depicts PIML-DCNN predictions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where P denotes the actual power output, P_{STC} is the power output under standard test conditions (STC), T_{mod} refers to the panel temperature from CFD results, $T_{\mathrm{STC}} = 25\,^{\circ}\mathrm{C}$ is the standard temperature, and $\beta_p = -0.45, \%/\mathrm{K}$ is the temperature coefficient of the panel [34].

Fig. 11 demonstrates that in arrays consisting of ten rows with staggered-height configurations, the first three rows exhibit improved power generation compared to those in uniform-height configurations. However, from the fourth row onward, there is a noticeable reduction in heat transfer efficiency. This decline in performance for panels located downstream is more pronounced in staggered-height configurations than in uniform-height scenarios.

3.4. Discussions

Although staggered-height configurations generally exhibit lower overall heat transfer performance compared to uniform-height configurations, the significantly enhanced heat transfer in the front rows of staggered-height arrays highlights a potential strategy. It suggests a balance could be achieved by employing staggered heights in the front rows while maintaining uniform heights in the rear rows. This approach could optimize the design of heterogeneous height arrangements for large-scale PV arrays. In such designs, the heights of PV panels are not simply varied in a periodic pattern but are strategically adjusted to maximize efficiency, with front rows in a staggered configuration and subsequent rows uniform. This configuration adds complexity to the design process and would likely increase costs if traditional experimental and computational fluid dynamics (CFD) methods are employed to quantify the convective heat transfer coefficient. Consequently, our hybrid learning model presents significant advantages, enabling rapid and accurate optimization of such configurations, which is particularly beneficial for industrial applications.

4. Conclusion

This work introduces an innovative approach that evaluates how the geometric layouts of PV modules influence their rate of convective heat loss under various environmental conditions. It achieves this by employing a combination of Physics-Informed Machine Learning (PIML) and a Deep Convolutional Neural Network (DCNN) to develop the PIML-DCNN approach. Additionally, we introduce a novel loss function termed "Pocket Loss", which enhances the interpretability of the generated physical parameters within the PIML-DCNN model. This function

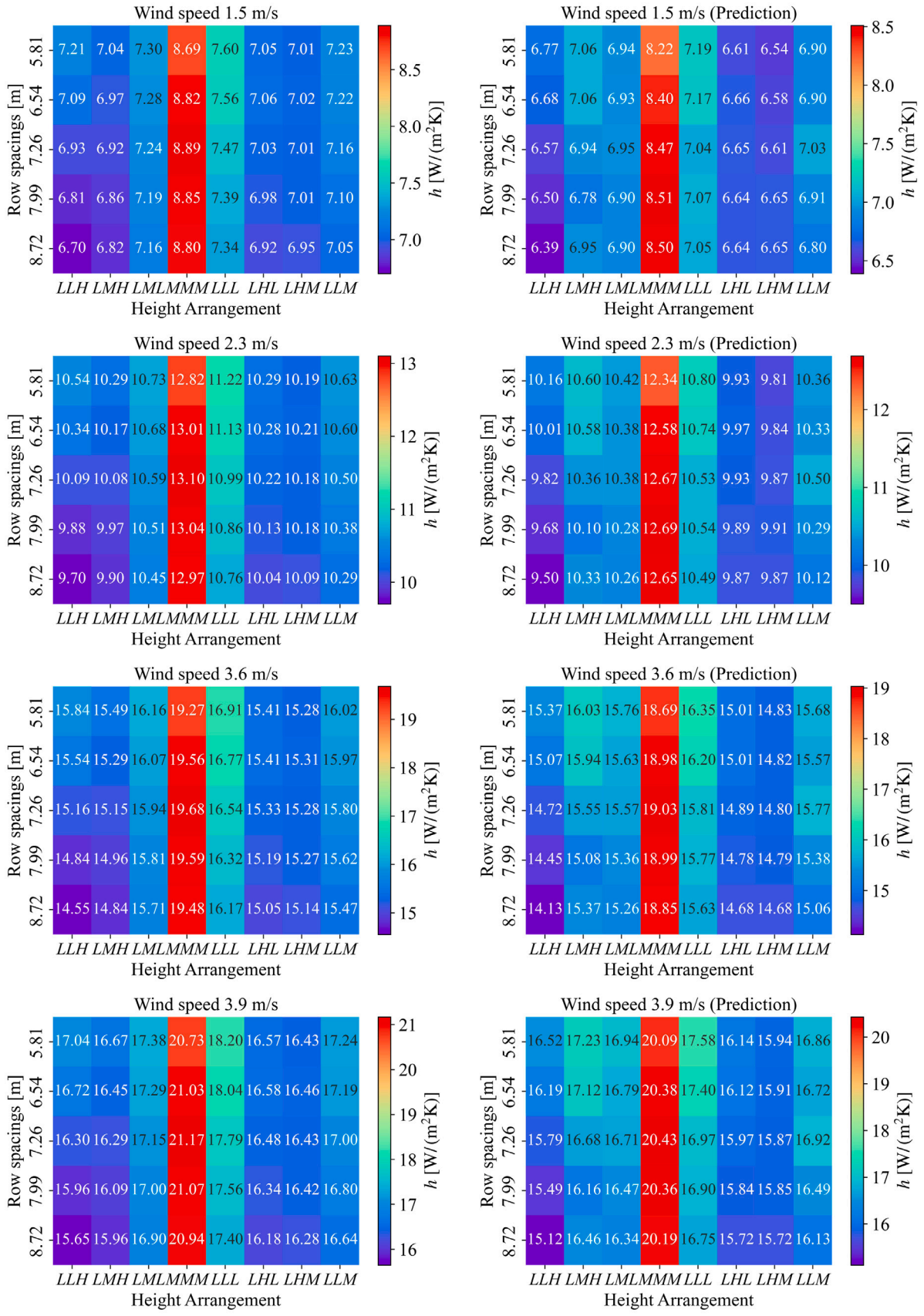


Fig. 10. Estimated h from (a) CFD method (left column) and (b) PIML-DCNN method (right column) for each array configuration for different wind velocities. The consistency in color distributions between the CFD and PIML-DCNN results across all cases indicates the accuracy of our model. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

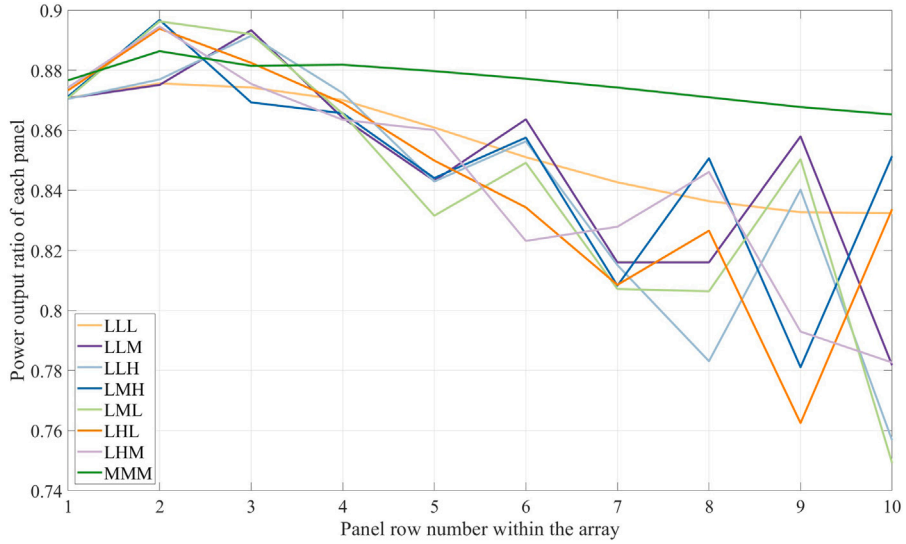


Fig. 11. Relative power output for each panel in arrays with diverse configurations. All cases maintain identical row spacing of 7.99 m and a wind velocity of 2.3 m/s. Other scenarios with different row spacings and wind velocities have comparable trends.

guides the model towards optimal results within a numerical range that adheres to physical principles. The developed PIML-DCNN model exhibits relative errors of 2.5% on the validation dataset and 2.7% on testing dataset when estimating the averaged convective heat transfer coefficient of the PV array, compared to validated CFD simulations. Once trained, the PIML-DCNN model can provide nearly instantaneous estimates of the convective heat transfer coefficient for new PV array configurations, striking an optimal balance between accuracy and computational efficiency. To investigate the influence of the array configuration on power output, we examined the power output of each panel within a ten-row array. The analysis indicates that in arrays with staggered heights, the first three rows outperform those in uniformly heightened arrays in terms of power generation. However, from the fourth row onward, a significant reduction in heat transfer efficiency becomes evident. Panels set at a uniform but elevated height across all rows demonstrate the most effective convective cooling performance. By capturing the impact of array layouts on convective heat dissipation, the proposed model offers an effective tool for optimizing PV array design, ultimately enhancing power generation efficiency in practical applications.

CRediT authorship contribution statement

Dapeng Wang: Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Zhaojian Liang:** Writing – review & editing. **Ziqi Zhang:** Writing – original draft. **Mengying Li:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used GPT-4 to improve the language. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

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

Acknowledgments

This work is substantially supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. C6003-22Y). This work is also supported by the Hong Kong Polytechnic University Undergraduate Research and Innovation Scheme (Project No. P0043659).

Appendix

Validation of the CFD model

The grid independence test of the CFD model is illustrated in Fig. A.1. When the number of mesh elements reaches 152,539, further refinement results in negligible differences in the calculated average surface temperature of the PV array. Consequently, this mesh density is utilized for all subsequent numerical experiments.

The CFD results are validated against the Large Eddy Simulation (LES) results reported in [11]. The fluid domain of the LES model was three-dimensional and was specifically configured to closely encompass the PV array. The span length of this domain matches the width of a single PV panel with dimensional ratios of $W/L = 0.01$ and $W/Z = 0.04$. Given that the variations in the physical fields along the spanwise dimension are small when compared to the other two dimensions, we propose simplifying the model by omitting the spanwise dimension. This simplification is intended to decrease computational costs while still preserving an acceptable level of accuracy in heat transfer predictions.

For the purpose of validation, we simulate scenarios analogous to those described in [11], where panels are maintained at a constant temperature of 320.15 K and a height of 1.52 m, and are spaced at various intervals under a wind speed of $U_\infty = 3.6$ m/s. The average convective heat transfer coefficient from our CFD simulations is calculated according to Eq. (1).

The Nusselt number (Nu) is employed as the primary metric for validation. The relative difference, e is defined as:

$$e = \frac{|\mathrm{Nu}_i - \widehat{\mathrm{Nu}_i}|}{\min(\mathrm{Nu}_i, \widehat{\mathrm{Nu}_i})} \times 100 \tag{16}$$

Here, Nu_i represents the Nusselt number derived from the benchmark large-eddy simulation (LES), and $\widehat{\mathrm{Nu}_i}$ denotes the average Nusselt

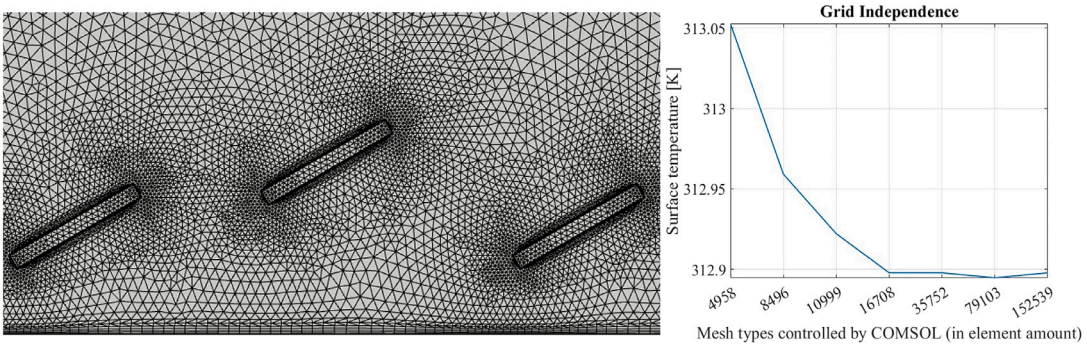


Fig. A.1. Grid independence test of the LML case with row spacing of 5.81 m and U_∞ of 3.6 m/s.

Table A.1
Validation of CFD results.

Row spacing [m]	Nu_i	$\widehat{Nu_i}$	e
5.81	1735.6	1621.4	7.0%
6.54	1679.2	1650.3	1.7%
7.26	1621.2	1650.3	1.8%
7.99	1578.9	1664.7	5.4%
8.72	1551.0	1679.2	8.3%

number obtained from our CFD model. In both approaches, the panel height of 1.52 m is utilized as the characteristic length for calculating Nu in the validation scenarios. As indicated in Table A.1, the relative difference for each case is maintained within 8.3%, confirming the accuracy and reliability of our CFD model.

Independence test of lacunarity calculation

The independence test for lacunarity calculation is conducted by systematically reducing the initial value of r_N, defined as half the minimum dimension of a PV panel, via an adaptive gradient method. In our experiments, the value of L_c remained stable even as r_N was reduced to half its previous value with each iteration, though this significantly increased the computation time. For instance, computing a single L_c value takes approximately 30 min when r_N is reduced to one-quarter of its initial size, utilizing a server equipped with an AMD Ryzen Threadripper 3970X 32-Core CPU.

To optimize the balance between efficiency and accuracy, we implemented an adaptive gradient mechanism as described in Eq. (17). This strategy permits r_N to initially increase from a low baseline, with a decelerating rate of change. In this context, i denotes the ith update of r_N, and g_0 is the predetermined stable gradient value during the test, set at 0.8 for our experiments. This method is designed to maintain both the accuracy and efficiency of the test.

$$g_i = \frac{g_0}{1 + e^{-i}} \tag{17}$$

Consequently, r_N updates as follows at each time step i:

$$r^i N = gi \times r_N^{i-1} \tag{18}$$

This strategy of initially decreasing r_N significantly and then gradually reducing the rate of decrease helps to determine a final value of r_N that does not undermine the accuracy of L_c quantification. This approach effectively balances precision with computational efficiency. One of the outcomes of the L_c independence test is illustrated in Fig. A.2.

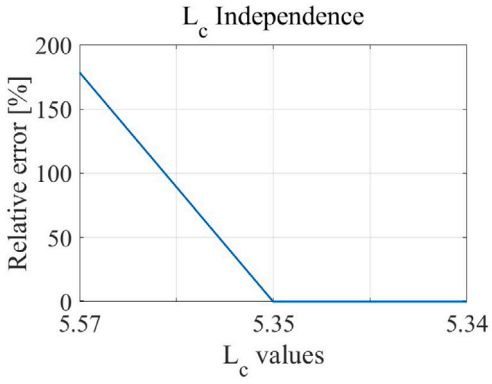


Fig. A.2. Independence test of L_c for LML configuration with row spacing 6.54 m.

Proof of the universal approximator

The linearization process within the PIML-DCNN model, particularly the feedforward propagation of the Reynolds number (Re) without utilizing an activation function, is concisely captured by the following equations. These equations illustrate the workings of a linear neural network:

$$y_1 = \mathbf{w}_1^{\mathrm{T}} \mathrm{Re} + b_1 \tag{19}$$

$$\mathrm{Nu} = \mathbf{w}_2^{\mathrm{T}} y_1 + b_2 \tag{20}$$

In these equations, $\mathbf{w}_i$ and b_i represent the weight matrix and bias vector of the ith layer, respectively, [29]. The first layer, named Dense4, and the second layer, named Dense5, function collaboratively to compute the Nusselt number (Nu) from Re. By integrating Eqs. (19) and (20), we derive the comprehensive formula used to calculate Nu:

$$\mathrm{Nu} = \mathbf{w}_2^{\mathrm{T}}(\mathbf{w}_1^{\mathrm{T}} \mathrm{Re} + b_1) + b_2 \tag{21}$$

Upon simplification, recognizing that the term $\mathbf{w}_2^{\mathrm{T}} b_1 + b_2$ constitutes a constant, Eq. (21) effectively represents an approximation of a nonlinear relationship involving Re. This approximation is detailed further in Eq. (10), demonstrating the model's capability to linearly approximate a nonlinear function through its structure.

Data availability

Data will be made available on request.

References

[1] Lamaamar I, Tilioua A, Hamdi Alaoui MA. Thermal performance analysis of a poly c-Si PV module under semi-arid conditions. Mater Sci Energy Technol 2022;5:243–51.

[2] Harmailil IO, Sultan SM, Tso C, Fudholi A, Mohammad M, Ibrahim A. A review on recent photovoltaic module cooling techniques: Types and assessment methods. Results Eng 2024;22:102225.

[3] Badi N, Laatar AH. Improved cooling of photovoltaic panels by natural convection flow in a channel with adiabatic extensions. PLOS ONE 2024;19(7):1–23.
[4] Glick A, Ali N, Bossuyt J, Calaf M, Cal RB. Utility-scale solar PV performance enhancements through system-level modifications. Sci Rep 2020;10.
[5] Bilawane R, Mandavgade NK, Kalbande VN, Patle LJ, Kanojiya M, Khorgade RD. Experimental investigation of natural convection heat transfer coefficient for roughed inclined plate. Mater Today: Proc 2021;46:7926–31, 3rd International Conference on Materials, Manufacturing and Modelling.
[6] Pawar VR, Sobhansarbandi S. Design optimization and heat transfer enhancement of energy storage based solar thermal collector. Sustain Energy Technol Assess 2021;46:101260.
[7] Pretorius J, Nielsen S. Understanding heat dissipation factors for fixed-tilt and single-axis tracked open-rack photovoltaic modules: Experimental insights. Prog Photovolt, Res Appl 2024;n/a(n/a):18.
[8] Smith SE, Viggiano B, Ali N, Silverman TJ, Obligado M, Calaf M, Cal RB. Increased panel height enhances cooling for photovoltaic solar farms. Appl Energy 2022;325:119819.
[9] Kazemian A, Khosravi K, Sen S, Talebizadehsardari P, Cairns A, Ma T. Optimizing photovoltaic thermal systems with wavy collector tube: A response surface-based design study with desirability analysis. Appl Therm Eng 2025;258:124475.
[10] Glick A, Ali N, Bossuyt J, Recktenwald G, Calaf M, Cal RB. Infinite photovoltaic solar arrays: Considering flux of momentum and heat transfer. Renew Energy 2020;156:791–803.
[11] Stanislawski BJ, Harman T, Silverman TJ, Cal RB, Calaf M. Row spacing as a controller of solar module temperature and power output in solar farms. J Renew Sustain Energy 2022;14(6):063702.
[12] Jaffer A. Natural convection heat transfer from an isothermal plate. Thermo 2023;3(1):148–75.
[13] Smith SE, Stanislawski BJ, Eng BK, Ali N, Silverman TJ, Calaf M, Cal RB. Viewing convection as a solar farm phenomenon broadens modern power predictions for solar photovoltaics. J Renew Sustain Energy 2022;14(6):063502.
[14] Wei H, Bao H, Ruan X. Perspective: Predicting and optimizing thermal transport properties with machine learning methods. Energy AI 2022;8:100153.
[15] Bhasme P, Vagadiya J, Bhatia U. Enhancing predictive skills in physically-consistent way: Physics informed machine learning for hydrological processes. J Hydrol 2022;615:128618.
[16] Casini M, De Angelis P, Chiavazzo E, Bergamasco L. Current trends on the use of deep learning methods for image analysis in energy applications. Energy AI 2024;15:100330.
[17] Efatinasab E, Irannezhad N, Rampazzo M, Diani A. Machine and deep learning driven models for the design of heat exchangers with micro-finned tubes. Energy AI 2024;16:100370.
[18] Hughes R, Haidinger T, Pei X, Vagg C. Real-time temperature prediction of electric machines using machine learning with physically informed features. Energy AI 2023;14:100288.
[19] Kim H, Kim J, Lee C. Interpretable deep learning for prediction of Prandtl number effect in turbulent heat transfer. J Fluid Mech 2023;955:A14.
[20] Yang Z, Gaidhane AD, Drgoňa J, Chandan V, Halappanavar MM, Liu F, Cao Y. Physics-constrained graph modeling for building thermal dynamics. Energy AI 2024;16:100346.
[21] Kalpana V, Jessy Sujana G, Thyagarajan K, Lalitha R, Talasila V, Mohan Jadhav M. Enhancing heat transfer coefficient predictions in complex geometries through hybrid machine learning approaches. Therm Sci Eng Prog 2024;55:103017.
[22] Chu Y, Wang Y, Yang D, Chen S, Li M. A review of distributed solar forecasting with remote sensing and deep learning. Renew Sustain Energy Rev 2024;198:114391.
[23] Chen S, Li C, Stull R, Li M. Improved satellite-based intra-day solar forecasting with a chain of deep learning models. Energy Convers Manage 2024;313:118598.
[24] Jing T, Chen S, Navarro-Alarcon D, Chu Y, Li M. SolarFusionNet: Enhanced solar irradiance forecasting via automated multi-modal feature selection and cross-modal fusion. IEEE Trans Sustain Energy 2024.
[25] Sparrow E, Tien K. Forced convection heat transfer at an inclined and yawed square plate-application to solar collectors. J Heat Transf 1977;99(4):507–12.
[26] El-shamy A, Sakr RY, Berbish NS. Experimental and numerical study of forced convection heat transfer from an inclined heated plate placed beneath a porous medium. Al- Azhar Univ Eng J 2007;2(4).
[27] Bergman TL. Fundamentals of heat and mass transfer. John Wiley & Sons; 2011.
[28] Luo Y, Wang M, Zhou H, Yao Q, Tu W-W, Chen Y, Dai W, Yang Q. AutoCross: Automatic feature crossing for tabular data in real-world applications. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. KDD '19, New York, NY, USA: Association for Computing Machinery; 2019, p. 1936–45.
[29] Svozil D, Kvasnicka V, Pospichal J. Introduction to multi-layer feed-forward neural networks. Chemometr Intell Lab Syst 1997;39(1):43–62.
[30] Siddique N, Paheding S, Elkin CP, Devabhaktuni V. U-Net and its variants for medical image segmentation: A review of theory and applications. IEEE Access 2021;9:82031–57.
[31] Zhou T, Zhang X, Droguett EL, Mosleh A. A generic physics-informed neural network-based framework for reliability assessment of multi-state systems. Reliab Eng Syst Saf 2023;229:108835.
[32] Rao C, Sun H, Liu Y. Physics-informed deep learning for incompressible laminar flows. Theor Appl Mech Lett 2020;10(3):207–12.
[33] Deng F, Tao X, Wei P, Wei S. A robust deep learning-based damage identification approach for SHM considering missing data. Appl Sci 2023;13(9).
[34] Vaillon R, Dupré O, Cal RB, Calaf M. Pathways for mitigating thermal losses in solar photovoltaics. Sci Rep 2018;8(1):13163.