Structures 79 (2025) 109529

Contents lists available at ScienceDirect

Structures

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

Hybrid neural network methods to model the external wind pressure on a low-rise flat-roofed building in an irregularly shaped urban environment

Hirushan Sajindra [a], Shashika Dharmawansha [b], Hasanthi Wijesundara [c], Sumudu Herath [b], Upaka Rathnayake [d], D.P.P. Meddage [e,f,*]

[a] Department of Computer Systems Engineering, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka
[b] Department of Civil Engineering, University of Moratuwa, Sri Lanka
[c] Department of Civil Engineering, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka
[d] Department of Civil and Environmental Engineering, Atlantic Technological University, Sligo, Ireland
[e] Ceylon Institute for Artificial Intelligence and Research (CIAIR), Colombo, Sri Lanka
[f] School of Engineering and Information Technology, The University of New South Wales, Canberra, ACT 2600, Australia

ARTICLE INFO

Keywords:
Neural networks
Wind pressure
Low-rise building
Urban environment
Wind tunnel

ABSTRACT

The present study used hybrid artificial neural networks to model the wind pressure (mean and fluctuating) on a flat-roofed, low-rise building in an irregularly shaped urban environment. Four neural networks, each combined with an artificial bee colony (ABC), genetic algorithm (GA), particle swarm optimisation (PSO), and independent component analysis (ICA), along with an individual artificial neural network (ANN) model and a convolutional neural network (CNN), were used for the wind pressure predictions. The data was obtained from Tokyo Polytechnic University's boundary layer wind tunnel and was used to train the neural network models. The results revealed that all models accurately captured the wind pressure on the low-rise building in a dense urban environment. Specifically, the genetic algorithm-artificial neural network (GA-ANN) model outperformed the remaining models, achieving good prediction accuracy for test data (coefficient of determination (R^2) = 0.96 for mean pressure $R^2 = 0.84$ for fluctuation pressure). The use of machine learning explainability methods confirmed the consistency of GA-ANN with the fundamentals of wind engineering. Notably, the GA-ANN approach accurately modeled the special flow features on the building surface, such as flow separation, vortex formation, and pressure gradients, to a greater extent compared to the wind tunnel results. Therefore, the authors propose this method as an complementary approach for predicting wind pressure on low-rise buildings in complex urban environments.

1. Introduction

Low-rise buildings are generally considered to be those with a height of less than 18 m and with the least horizontal dimension greater than their height, according to the American Society of Civil Engineers [1]. These buildings are immersed in the lowest part of the atmospheric boundary layer, where turbulence is typically high. This high turbulence creates a highly fluctuating nature in wind flow, resulting in complex wind pressure distributions on low-rise buildings. Proper estimation of these wind loads is crucial for maintaining the safety and stability of low-rise buildings. However, most low-rise buildings are non-engineered and rarely account for wind loads on the building envelope during design [1]. Therefore, accurate estimation of wind loads is essential for the safe and economical design of a low-rise building.

Experimental methods, such as full-scale or wind tunnel experiments, have been conducted for several decades to investigate wind pressure on low-rise buildings [2]. These experiments are the most reliable method for examining wind pressure distributions on such buildings. It is noteworthy that, over the years, a large amount of data from wind tunnel experiments on low-rise buildings has become available. The majority of these experiments were conducted by Holmes [2,3], Davenport et al. [4], Stathopoulos [5], Stathopoulos & Luchian [6], Stathopoulos & Saathoff [7] to develop provisions in international wind loading standards. However, wind tunnel or full-scale experiments come with certain constraints: experiments require significant effort, are time-consuming, require expertise, and are costly [8]. Additionally,

* Corresponding author at: School of Engineering and Information Technology, The University of New South Wales, Canberra, ACT 2600, Australia.
E-mail address: p.meddage@unsw.edu.au (D.P.P. Meddage).

https://doi.org/10.1016/j.istruc.2025.109529
Received 1 January 2025; Received in revised form 25 March 2025; Accepted 20 June 2025
Available online 23 June 2025

wind tunnels or full-scale testing facilities are not commonly available to engineering practitioners. To mitigate these constraints to a certain extent, numerical modelling was introduced in the form of computational fluid dynamics (CFD). CFD modelling of wind behavior on low-rise buildings has become very popular in the wind engineering domain [9]. Once a numerical model is properly validated using experimental data, it can be used to change the building's geometry or surrounding environment and simulate wind flow [9]. This approach can lower the cost and physical effort required in experimental modelling. However, numerical modelling requires high computational power to model urban-like settings around low-rise buildings. Moreover, it also requires expertise to develop a high-quality computational mesh (grid) in the model and domain. If the mesh structure becomes complex, including additional details around the building, solving it without high-end computational resources becomes difficult.

Modelling wind pressure in urban settings is challenging using both experimental and numerical methods, as previously discussed. An alternative approach involves data-driven methods such as machine learning, which can effectively model wind behavior on low-rise buildings using existing data [8]. Machine learning requires lower computational costs compared to traditional numerical modelling of wind pressure on buildings. Once trained, a machine learning model can quickly provide repetitive predictions within short time frames [10]. Moreover, machine learning can uncover hidden patterns in wind pressure data, identifying areas prone to higher wind loads on low-rise buildings. It processes vast amounts of data swiftly (typically within minutes or hours) and delivers accurate predictions, making it advantageous for engineers. In contrast, numerical modelling involves repetitive simulations, each potentially lasting from hours to days depending on the complexity of the building's surroundings.

Machine learning has become prevalent in wind engineering research, particularly concerning low-rise buildings. For instance, Chen et al. [11] utilised artificial neural networks (ANNs) to predict wind pressure time histories on a roof panel of a low-rise building measuring 30 m x 45 m with a height of 9.75 m. Additionally, Chen et al. [12] employed neural networks to model static pressure on low-rise buildings using data from wind tunnel tests conducted at a 1:100 scale. Their model obtained good predictions with average mean square values of 8.9 and 11.6. Vrachimi et al. [13] developed an artificial neural network to predict the static pressure on a low-rise, flat-roofed building with a 16 m x 16 m plan area. They reported that using artificial neural networks can improve the predictions of local wind pressure coefficients. Ma et al. [14] used a neural network and Gaussian process regression to interpolate surface wind pressure using wind tunnel data from a low-rise building. The Gaussian process regression model achieved a coefficient of correlation of 0.91–0.93 for surface pressure interpolation and 0.89–0.96 for pressure extrapolation.

Compared to neural networks, classical machine learning approaches like tree-based methods have also been adopted in wind engineering research. For example, Hu et al. [15] used decision trees, random forests, extreme gradient boosting, and a generative adversarial network to predict static wind pressure on a high-rise building with interference effects. They observed that the generative adversarial network obtained the best predictions ($R^2 = 0.988$ for mean pressure predictions and $R^2 = 0.924$ for fluctuation pressure predictions). However, their tree-based methods also achieved comparable prediction accuracy. Recent work by Huang et al. [16] applied deep neural networks, convolutional neural networks, and artificial neural networks to model static wind pressure on low-rise buildings with gable roofs. The convolutional neural network method looked promising and achieved a relative error of 0.1 % for the predictions obtained in roof corners where intense peak suction pressure is expected.

It is important to note that most of these studies have focused on low-rise buildings in isolated environments. Predicting wind pressure on low-rise buildings amidst surrounding structures is complex, yet crucial for accurate assessment of wind loads. Limited studies in wind engineering have utilised machine learning for this purpose. For example, Meddage et al. [8] utilised tree-based models to predict external wind pressure coefficients on a low-rise building in a regular urban setting. Their best-performing model, extreme gradient boosting, achieved an R^2 of 0.94 for mean and fluctuation wind pressure predictions. However, their study did not include urban environments where buildings are irregularly/randomly arranged. Precise estimation of wind pressure on low-rise buildings in such complex urban configurations is essential to safeguard them against severe wind events. No study has used machine learning methods to predict the wind pressure characteristics on a low-rise building surrounded by similar buildings in a random or irregular arrangement.

Recognising this gap, we employed a machine learning approach to predict wind pressure on a low-rise flat-roofed building in an irregularly shaped urban environment. Given that the extreme gradient boosting model used by Meddage et al. [8] achieved an R^2 of only 0.94 for predictions, we explored the potential for improved accuracy using hybrid neural network methods. This reason for using hybrid methods was inspired by the recent research progress in engineering observed with hybrid neural networks [17,18]. Four neural networks, each incorporating an artificial bee colony (ABC), genetic algorithm (GA), particle swarm optimisation (PSO), and independent component analysis (ICA), in addition to an individual artificial neural network model (ANN) and a convolutional neural network (CNN), were utilised as machine learning methods. Those were trained to predict the mean wind pressure and fluctuating wind pressure on a low-rise flat-roofed building surrounded by similar buildings in a random arrangement.

Furthermore, Shapley additive explanations (SHAP) and local interpretable model agnostic explanations (LIME) were used to obtain the feature importance of predictions to confirm that the predictions adhere to the fundamentals of wind engineering. SHAP and LIME have been used very rarely in wind engineering [8,19,20] and those studies highlighted that these methods are useful in elucidating the relationship between the wind pressure, geometric parameters, and wind direction.

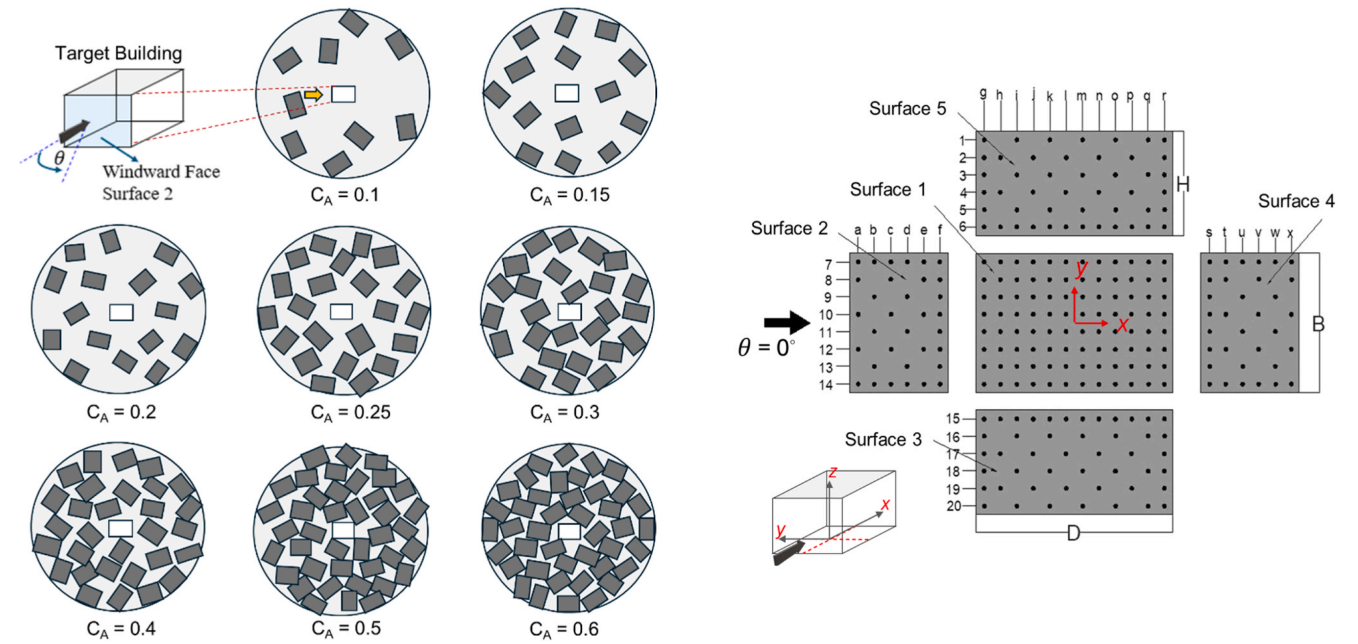
To our knowledge, this study represents the first of its kind to employ hybrid optimised neural network techniques for predicting wind pressure on low-rise buildings situated in highly complex urban environments with randomly/irregularly arranged surrounding buildings. Wind tunnel data from the Tokyo Polytechnic University (TPU) database was utilised to train the neural network models [21]. The following section details the test configurations and data employed in this investigation.

2. Wind tunnel data

The neural network models were developed using wind tunnel data to predict mean and fluctuating pressures on non-isolated low-rise buildings. Wind pressure on such buildings can be more complex compared to isolated structures, influenced by factors like geometric parameters, surrounding building density, wind incident angle, and approaching turbulence. Notably, the presence of nearby buildings can significantly alter wind pressures on the target building. Therefore, predicting wind pressures accurately in real urban environments becomes challenging due to the interaction of these factors. Recent wind engineering studies emphasise the importance of understanding wind pressure distributions in urban areas for natural ventilation [22], facade and cladding wind loading [23], and pollutant dispersion [24].

2.1. Building model and wind tunnel setup

The present study utilised external wind pressure coefficients on a flat-roofed low-rise building surrounded by randomly arranged low-rise buildings, sourced from the Tokyo Polytechnic University wind tunnel database [21]. The target building selected in this study was flat-roofed and shared similar geometrical dimensions with the surrounding buildings. Under the random arrangement of non-isolated low-rise flat-roofed buildings, the dataset included eight different densities of



(a) Surrounding building arrangement (b) Pressure tap configuration on the target building

Fig. 1. Experimental setup in the wind tunnel test (θ: wind direction, C_A: density of surrounding buildings and the vertical direction is the *z-axis*).

Table 1
Calculation of C_A values based on the area of surrounding of buildings.

C_A provided in the TPU database	Number of surrounding buildings in the Random arrangement	C_A Calculated	C_A Used for the study
0.1	9	0.12	0.1
0.15	14	0.18	0.15
0.2	16	0.22	0.2
0.25	22	0.30	**0.3**
0.3	26	0.35	**0.35**
0.4	30	0.41	0.4
0.5	40	0.52	0.5
0.6	42	0.58	0.6

surrounding buildings (C_A) ranging from 0.1 to 0.6 (0.1, 0.15, 0.2, 0.3, 0.35, 0.4, 0.5, and 0.6), where C_A represents the ratio of the area occupied by all buildings to the total site area (Fig. 1.a). Table 1 presents the calculation of C_A values. It is noteworthy that the C_A calculated for random/irregular arrangement slightly deviated from the C_A values given in the database. Therefore, when the deviation was significant, the calculated values were used.

The target and surrounding buildings were scaled down at a ratio of 1:100 for wind tunnel testing with a velocity scale of 1:3. Each building model, including the target, measured 240 mm (depth) x 160 mm (breadth) x 120 mm (height).

The target building was subjected to testing under nine wind directions (0°, 23°, 45°, 68°, 90°, 113°, 135°, 158°, and 180°), resulting in 72 different cases (8 densities x 9 directions) in the dataset. Each case involved the target building model with 256 pressure taps covering its surface, as illustrated in Fig. 1.b. The experiments were conducted in a boundary layer wind tunnel with suburban terrain (Terrain category III) conditions according to AIJ 2004 standard [25], at a wind speed of 7.8 m/s measured at a height of 100 mm above the wind tunnel bed (Corresponds to 23.4 m/s at a height of 10 m in full scale). A turbulence intensity of 25 % was measured at a height of 100 mm in the wind tunnel with a decaying pattern with height. Instantaneous pressure recordings on the target building were sampled at 781.25 Hz for 18 s.

At each pressure tap location, time histories of pressure were recorded, from which mean pressure and fluctuation pressure coefficients were calculated using Eqs. 1, 2 and 3. In Eq. 1, p_i refers to the pressure measured at a point, p_o refers to the static pressure measured at a reference height, ρ is the density of air which is 1.2 kg/m^3. U_H is the velocity measured at reference height. $C_{p,i}$ refers to the instantaneous pressure coefficient. Mean pressure coefficients ($C_{p,mean}$) represent time-averaged flow patterns on the building, while higher fluctuation pressure coefficients ($C_{p,rms}$) indicate areas prone to high turbulence and

Table 2
Summary of the descriptive statistics of data.

	Feature	*Mean*	*Standard deviation*	*Minimum*	*Maximum*	*Skewness*
Inputs	Wind Angle (θ)	90.2	58	0	180	−0.003
	x	0	8.6	−12	12	0
	y	0	6.1	−8	8	0
	z	8.5	3.8	1	12	−0.74
	Surface	2.6	1.53	1	5	0.34
	Density (C_A)	0.33	0.17	0.1	0.6	0.28
Outputs	$C_{p,mean}$	−0.19	0.20	−1.49	0.72	0.012
	$C_{p,rms}$	0.16	0.07	0.08	0.84	1.94

**$C_{p,mean}$: mean pressure coefficient; $C_{p,rms}$: fluctuation pressure coefficient (rms: root mean square)

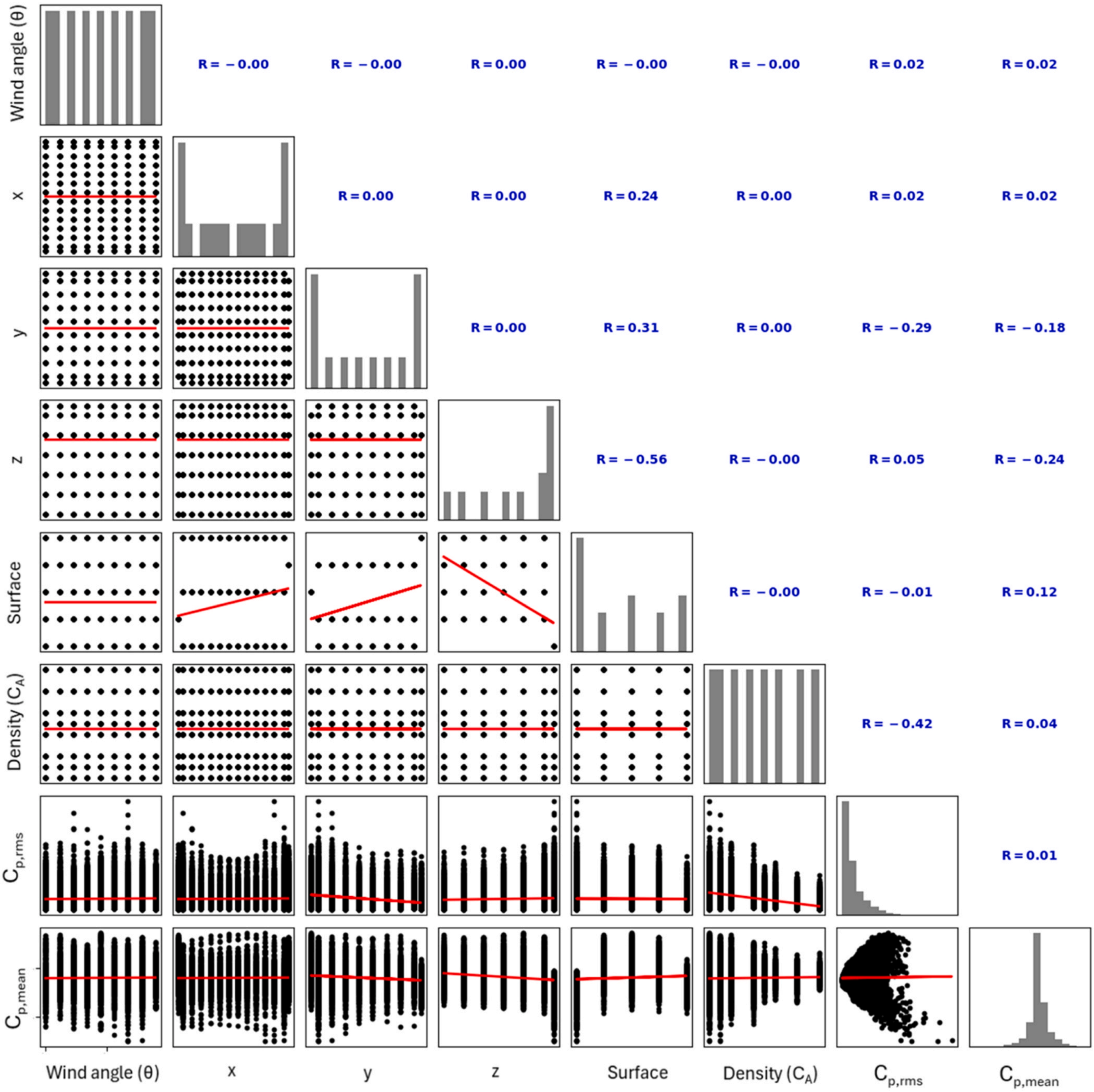


Fig. 2. Pairwise correlation matrix.

fluctuations. These areas typically experience intense localised wind pressures, particularly near the roof, leading edges, and corners, where high negative pressures can induce significant wind-induced uplift forces.

$$C_{p,i} = \frac{p_i - p_o}{0.5\rho U_H^2} \tag{1}$$

$$C_{p,mean} = \frac{1}{N}\sum_{i=1}^{n} C_{p,i} \tag{2}$$

$$C_{p,rms} = \sqrt{\frac{\sum_{i=1}^{n} (C_{p,i} - C_{p,mean})^2}{N}} \tag{3}$$

The dataset comprised 18,432 unique instances (72 cases x 256 pressure taps) with inputs including x, y, and z coordinates of pressure taps, surface of the building, wind direction, and surrounding building density (C_A). Table 2 presents the statistics of the dataset. These input parameters represent essential geometric parameters and wind direction parameters used by Meddage et al. [8]. Since their results showcase certain deviations in the predictions with respect to the original data, the current study used the building surface as an additional variable in the hybrid model. The reason is that each of these surfaces has unique wind flow characteristics, and the surface variable is used to denote that uniqueness.

According to Table 2, the parameters x and y had zero mean and skewness as the origin is located in the center of the target building at the ground level as shown in Fig. 1.b. The data distribution of $C_{p,rms}$ was

Table 3
Time taken for each model for optimisation, training and testing.

$C_{p,mean}$						
Time	ABC-ANN	GA-ANN	PSO-ANN	ICA-ANN	ANN	CNN
Optimisation	1 h 15 min	1 h 10 min	1 h 3 min	5 min	25 h	22 h
Training	28 min	35 min	32 min	12 min	5 min	4 min
Testing	15 s	17 s	15 s	10 s	1 s	1 s
$C_{p,rms}$						
Time	ABC-ANN	GA-ANN	PSO-ANN	ICA-ANN	ANN	CNN
Optimisation	1 h 30 min	1 h 25 min	1 h 17 min	11 min	23 h	21 h
Training	25 min	40 min	30 min	15 min	5 min	5 min
Testing	13 s	20 s	18 s	15 s	1 s	1 s

relatively skewed and the $C_{p,mean}$ had a symmetric distribution confirmed by the skewness parameter of 0.012. The surrounding building density changed from 0.1 to 0.6, ranging from a low-density arrangement to a dense and complex urban arrangement. These densities represent real-life building arrangements in urban environments. In order to analyse the linearity between the features, the pairwise correlation was employed. Fig. 2 illustrates the correlation matrix of the dataset. $C_{p,mean}$ exhibited a negative correlation with the y and z variables, with coefficients of −0.18 and −0.24, respectively.

Conversely, the variables wind angle, x, surface, and density display weak positive correlations with $C_{p,mean}$. $C_{p,rms}$ shows a negative correlation with the y and density (C_A), with coefficients of −0.29 and −0.42, respectively. The variables wind angle, x, z, and surface exhibit weak positive correlations with $C_{p,rms}$. These correlations indicate that the relationship between the geometric parameters, wind directions and density of surrounding buildings and wind pressure is nonlinear, suggesting the need for an advanced data-driven approach. Therefore, this study proposes the use of hybrid optimised artificial neural networks to predict mean and fluctuating pressures on a flat-roofed low-rise building in a dense urban environment.

3. Machine learning methods

3.1. Hybrid artificial neural network approach

The authors opted for hybrid artificial neural networks with individual neural networks to model the wind pressure on the low-rise building with surrounding interference. The choice of model was inspired by two reasons: (a) classical machine learning methods have underperformed in wind engineering analysis, and (b) hybrid neural network methods have shown good accuracies in various other engineering studies compared to individual neural network models [17,18]. As individual neural network models, an ANN and CNN models were used. These two models were compiled using Keras and Tensorflow [26] libraries whereas two models were optimised using a grid search method [27]. For the hybrid approach, multilayer perceptron (MLP) neural network model was used and combined using artificial bee colony (ABC), particle swarm optimisation (PSO), genetic algorithm (GA), and individual component analysis (ICA). The reason for using MLP in the hybrid models was its greater flexibility for hybrid methods. These four methods were selected as they are state-of-the-art methods used in many engineering studies [17,18]. Subsequently, four hybrid models were developed to predict $C_{p,mean}$ and $C_{p,rms}$ separately. The following sections describe the basic concepts in each method.

3.1.1. Artificial bee colony-artificial neural network (ABC-ANN)

The Artificial Bee Colony algorithm is a bio-inspired swarm-based method that works as a swarm of bees searching for the best nectar source, where the best source represents the optimal configuration in the neural network. The ABC algorithm operates outside the neural network but works closely with it to find the best configuration [28]. In the initial stage, an ABC wrapper function was created to serve as a fitness test for the MLP neural network. This function adjusts the size of the hidden layers based on the parameters provided by the ABC algorithm. The developed neural network is then trained using these new sizes, and the function calculates the mean square error (MSE) to measure the model's performance. The optimisation process begins by generating random combinations of hidden layer sizes, and neurons within the specified bounds and evaluating them using the ABC wrapper function. Each bee in the swarm tries a different combination, trains the neural network

Table 4
Optimum models and parameters to predict $C_{p,mean}$ (MSE: Mean squared error).

Parameters	ABC-ANN	GA-ANN	PSO-ANN	ICA -ANN	ANN	CNN
A: Optimum Neuron Architecture	80, 62, 76	76, 92, 118, 103, 106	78, 64, 76, 79, 70	80 × 5 (This is a feature extraction-based model)	60 × 5	64 × 4
A: Optimised from (ANN, and CNN were considered to have similar neurons in each layer)	[50,200]	[50,200]	[50,200]		[50,200] in 10	[40,120] in 4
B: No. of hidden layers	3	5	5	5	5	4
B: Optimised from	[3–6]	[3–6]	[3–6]	N/A	[3–6]	[3–6]
C: Learning rate	0.2	0.2	0.2	0.2	0.15	0.1
C: Optimised from	-	-	-	-	[0.1,0.4]	[0.1,0.4]
Kernel and Filter size	-	-	-	-	-	3 & 48
Flattening layer	-	-	-	-	-	1 × 192
Dropout rate	-	-	-	-	0.3	0.2
Population size	50	50	50	-	-	-
wmax	-	-	0.9	-	-	-
wmin	-	-	0.4	-	-	-
c1	-	-	1.5	-	-	-
c2	-	-	2.5	-	-	-
Max. no. of iterations	200	280	100	1000	150	200
Limit parameter	100	-	-	-	-	-
No. of independent components	-	-	-	9	-	-
Tolerance	-	-	-	0.1	-	-
Mutation percentage	-	0.5	-	-	-	-
Mutation probability	-	0.01	-	-	-	-
Gamma	-	0.5	-	-	-	-
Sigma	-	0.7	-	-	-	-
Cost function	MSE	MSE	MSE	MSE	MSE	MSE

Table 5
Optimum models and parameters to predict $C_{p,rms}$ (MSE: Mean squared error).

Parameters	ABC-ANN	GA-ANN	PSO-ANN	ICA -ANN	ANN	CNN
A: Optimum Neuron Architecture	62, 62, 80	59, 100, 57, 58, 89	78, 62, 76, 73, 77	80 × 5 (This is a feature extraction-based model)	80x4	72x4
A: Optimised from (ANN, and CNN were considered to have similar neurons in each layer)	[50,200]	[50,200]	[50,200]		[40,200] in 10	[40,120] in 4
B: No. of hidden layers	3	5	5	5	4	4
B: Optimised from	[3–6]	[3–6]	[3–6]	N/A	[3–6]	[3–6]
C: Learning rate	0.2	0.2	0.2	0.2	0.23	0.17
C: Optimised from	-	-	-	-	[0.1,0.4]	[0.1,0.4]
Kernel and Filter size	-	-	-	-	-	3 & 48
Flattening layer size	-	-	-	-	-	1 × 192
Dropout rate	-	-	-	-	0.3	0.2
Max. no. of iterations	200	321	113	1000	150	200
Population size	50	50	50	-	-	-
wmax	-	-	0.9	-	-	-
wmin	-	-	0.4	-	-	-
c1	-	-	1.5	-	-	-
c2	-	-	2.5	-	-	-
Limit parameter	100	-	-	-	-	-
No. of independent components	-	-	-	9	-	-
Tolerance	-	-	-	0.2	-	-
Mutation percentage	-	0.5	-	-	-	-
Mutation probability	-	0.01	-	-	-	-
Gamma	-	0.5	-	-	-	-
Sigma	-	0.7	-	-	-	-
Cost function	MSE	MSE	MSE	MSE	MSE	MSE

with those sizes, and calculates the mean squared error (MSE). This optimisation process continues with 5-fold cross validation with the bees exploring different combinations and refining their search based on the best solutions found so far. Once the optimisation is complete (refer to Table 3), the ABC algorithm returns the best parameters ((number of hidden layers, and neurons in each layer) it found (refer to Tables 4 and 5). Then, the hidden layer sizes of the neural network are updated according to the suggested values. After the updating process, the neural network is trained again with the optimised parameters. Finally, the evaluation process is performed using model performance indices for the training (with 5-Fold cross validation) and testing stages separately [29].

3.1.2. Genetic algorithm-artificial neural network (GA-ANN)

GA is a bio-inspired evolutionary method to discover the best values for a function [30]. Khandelwal and Armaghani [31] demonstrated that genetic algorithms can optimise the neural networks' connection weights and biases more effectively than random generation. It has several principles such as genetics, mutation, natural selection, and crossover. Hence, the genetic algorithm is regarded as a robust and comprehensive search algorithm. The algorithm begins with the population organisation stage, where each individual represents a possible solution to the problem. The GA operates outside the neural network but interacts with it to find the best configuration.

First, an evaluate_model function is created to act as a fitness test for the neural network. This function builds an ANN model with the given hyperparameters, trains it, and evaluates its accuracy. The GA starts by generating an initial population with random hyperparameters. Each individual is evaluated using the evaluate_model function, which trains the neural network and calculates its accuracy. The GA then uses selection, crossover, and mutation to create a new generation. Selection chooses the best-performing individuals, crossover combines genes to create new combinations, and mutation introduces random changes. This cycle of evaluation, selection, crossover, and mutation continues for the specified number of generations. Over time, the population evolves, and the best hyperparameters (number of hidden layers, and neurons in each layer) emerge. The best individual from the final generation represents the optimal hyperparameters for the neural network (refer to Tables 4 and 5).

3.1.3. Particle swarm optimisation-artificial neural network (PSO-ANN)

In this method, particle swarm optimisation (PSO) was used to improve the neural network's performance. The PSO works with a group of particles (solutions) that move through the search space to find the best solution. Each particle adjusts its position based on its own experience and that of its neighbors, aiming to find the optimal set of parameters for the neural network [32].

The PSO_wrapper function was created to test the fitness of the neural network. This function adjusts the hidden layer sizes based on PSO-provided parameters, trains the neural network, and calculates the model's accuracy. The PSO optimisation process begins by generating random combinations of hidden layer sizes within the specified bounds and evaluating them using the PSO_wrapper function. Each particle tries a different combination, trains the neural network, and calculates the accuracy. The algorithm tracks the best combination with the highest accuracy. This process continues for the defined number of iterations, with particles refining their search based on the best solutions found so far. Once optimisation is complete, the PSO algorithm returns the parameters (number of hidden layers, and neurons in each layer) as shown in Tables 4 and 5.

3.1.4. Independent component analysis-artificial neural network (ICA-ANN)

Independent Component Analysis (ICA) was used to enhance the neural network's performance by using feature extraction before feeding the data into the network. ICA is effective in separating a multivariate signal into additive, independent components, potentially improving ANN performance by extracting the most informative features [33]. This process decomposes the data into independent components, which are then used as input features for the neural network. By reducing data dimensionality and isolating independent components, ICA captures the essential structure of the data. The training process involved adjusting neuron weights and biases to minimise errors between predicted and actual values. In this study, an MLP model consists of five hidden layers, with each hidden layer containing 80 neurons was used for ICA-ANN as this is a feature selection-based optimisation. The 5-fold cross validation approach was employed for the training process of this model. Additional parameters are provided in Tables 4 and 5 for the reference.

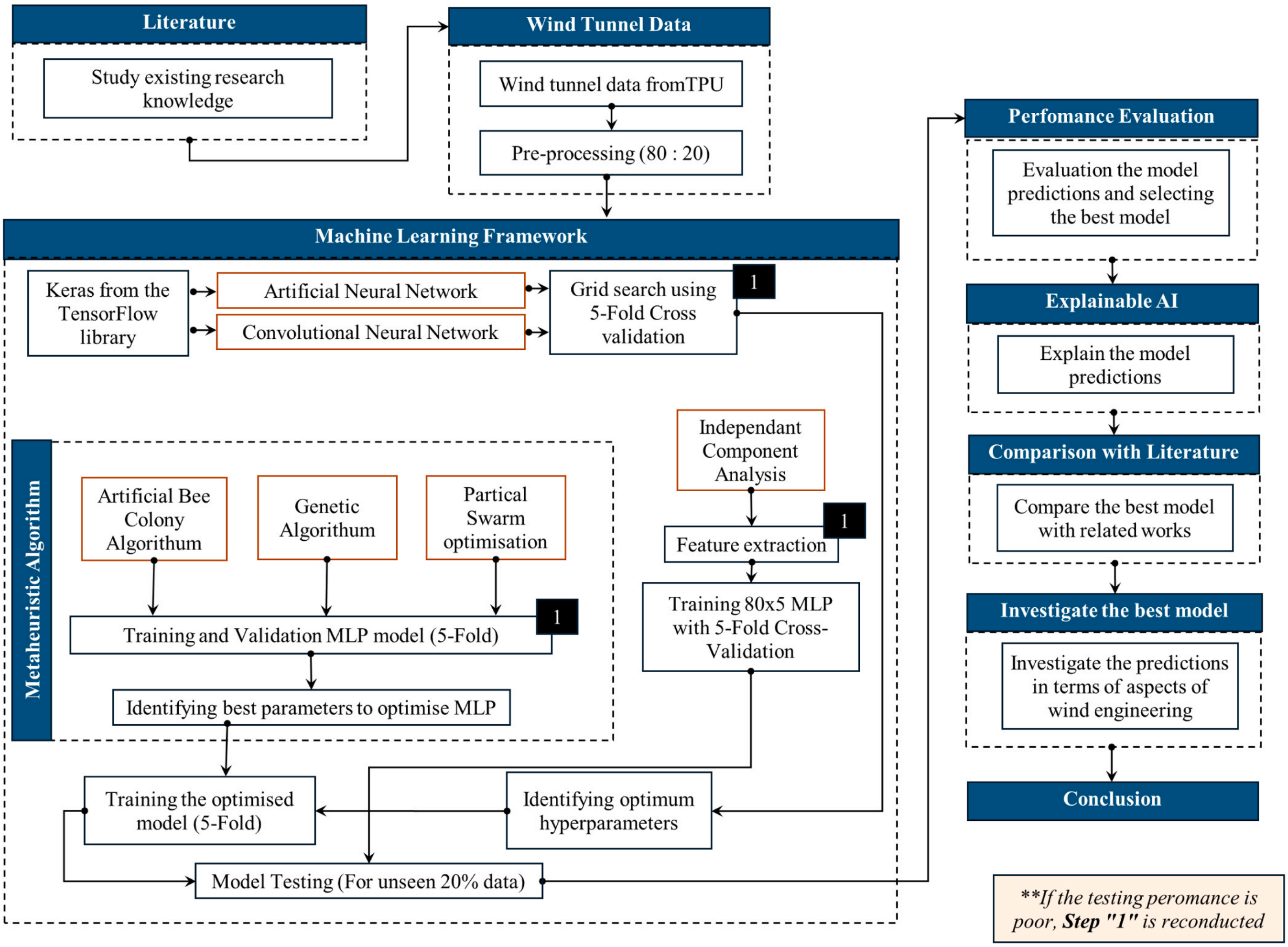


Fig. 3. Workflow of the study.

Table 6
Performance indices of machine learning models for $C_{p,mean}$ predictions.

$C_{p,mean}$	Model	ABC-ANN	GA-ANN	PSO-ANN	ICA-ANN	ANN	CNN
Training	R^2	0.884	0.976	0.921	0.948	0.913	0.913
	MSE	0.005	0.001	0.003	0.002	0.004	0.004
	RMSE	0.069	0.031	0.057	0.046	0.060	0.060
	MAE	0.048	0.022	0.040	0.033	0.040	0.041
Validation	R^2	0.863	0.963	0.889	0.916	0.866	0.883
	MSE	0.006	0.002	0.005	0.004	0.006	0.005
	RMSE	0.078	0.041	0.071	0.061	0.077	0.073
	MAE	0.054	0.027	0.048	0.040	0.051	0.047
Testing	R^2	0.846	0.959	0.880	0.904	0.853	0.870
	MSE	0.006	0.002	0.005	0.004	0.006	0.005
	RMSE	0.079	0.041	0.070	0.062	0.077	0.072
	MAE	0.054	0.027	0.046	0.040	0.049	0.047

3.1.5. *Artificial neural network (ANN)*

In this study, an individual ANN model was developed using the Keras and TensorFlow libraries. The ReLU activation function was utilised for the hidden layers due to its effectiveness in handling nonlinearity. During the compilation process, the Adam optimiser was used to optimise the model's parameters, and a grid search method was used to fine-tune the number of hidden layers, learning rate, and neurons in each layer. However, due to computational constraints, the authors considered a similar number of neurons in each layer during the grid search. A batch size of 500 was selected to balance computational efficiency during the training stage. To mitigate overfitting, a dropout rate of 0.3 was applied to the hidden layers. Optimum architecture of the models is provided in Tables 4 and 5.

3.1.6. *Convolutional neural network (CNN)*

The CNN model was also developed using the Keras and TensorFlow libraries. The model architecture consisted of 1D convolutional layer, with 48 filters and a kernel size of 3. The LeakyReLU activation function was used in the convolutional layers to introduce nonlinearity and improve feature extraction. A dropout rate of 0.2 was applied to mitigate overfitting, and the Adam optimiser was used. The flattening layer transformed the output of the convolutional layers into a 1×192 vector, which was then passed to a feed-forward network. A batch size of 500 was selected for the training process.

3.2. *Explainable artificial intelligence*

As many of these machine learning methods do not provide clear decision-making criteria for their predictions, the implementation of data-driven methods in wind engineering research remains challenging. Black-box predictions diminish the confidence of end users and domain experts. To address this, machine learning interpretability methods have been developed to explain the reasoning behind predictions. One such widely used interpretability method is Shapley Additive Explanations (SHAP), introduced by Lundberg and Lee [34]. SHAP has been successfully applied in the structural engineering domain, highlighting the importance of model explanations to aid end users in decision-making [35,36]. Moreover, local interpretable model agnostic explanations

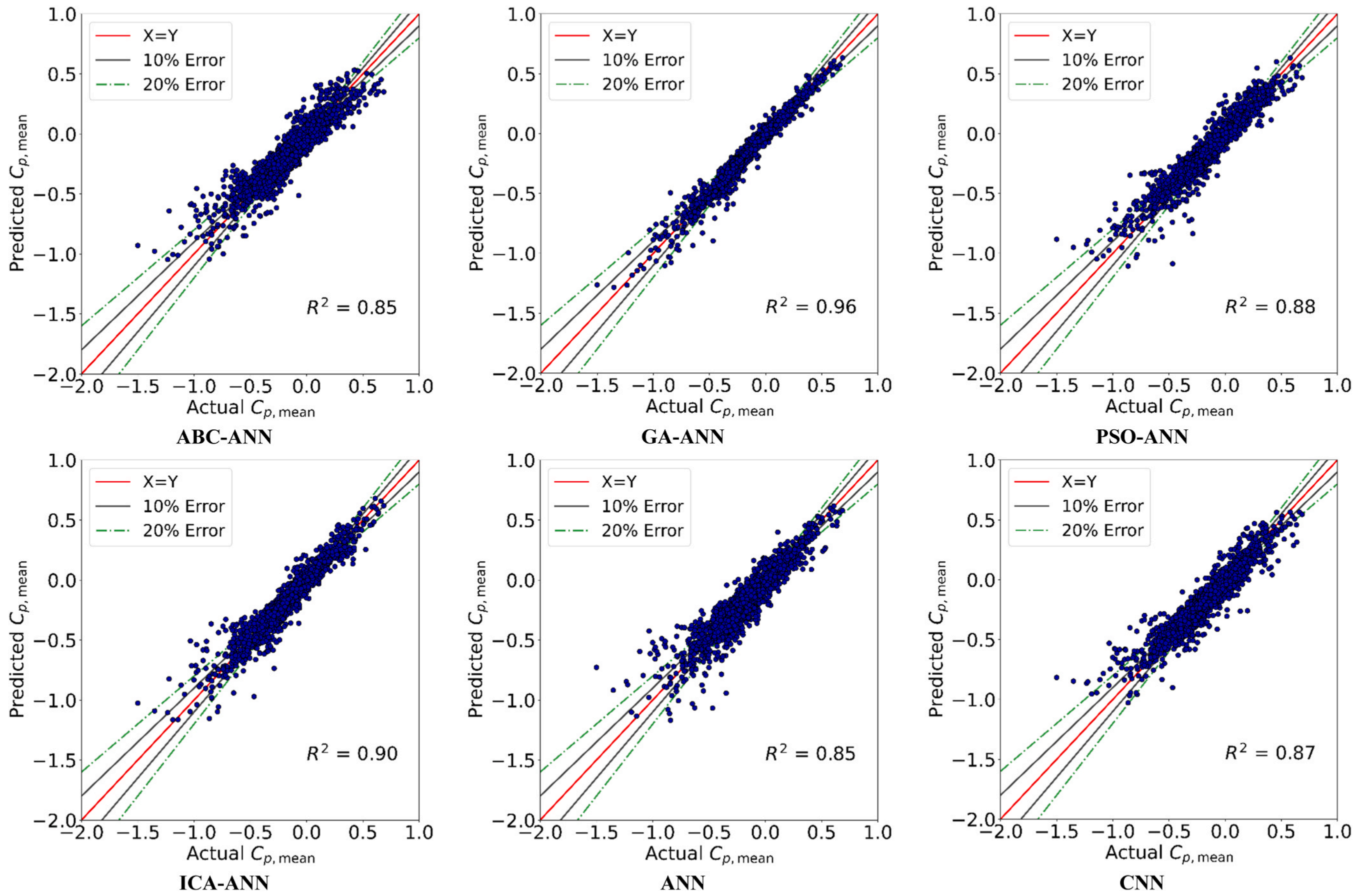


Fig. 4. Comparison of $C_{p,mean}$ predictions with the testing (wind tunnel data) $C_{p,mean}$.

Table 7
Performance indices of machine learning models for $C_{p,rms}$ predictions.

$C_{p,rms}$	Model	ABC-ANN	GA-ANN	PSO-ANN	ICA-ANN	ANN	CNN
	R^2	0.825	0.877	0.858	0.855	0.873	0.873
Training	MSE	0.001	0.001	0.001	0.001	0.001	0.001
	RMSE	0.031	0.026	0.028	0.028	0.026	0.026
	MAE	0.021	0.017	0.018	0.019	0.018	0.017
	R^2	0.825	0.869	0.847	0.842	0.836	0.863
Validation	MSE	0.001	0.001	0.001	0.001	0.001	0.001
	RMSE	0.031	0.027	0.029	0.029	0.030	0.027
	MAE	0.021	0.018	0.019	0.020	0.020	0.018
	R^2	0.790	0.839	0.810	0.810	0.796	0.833
Testing	MSE	0.001	0.001	0.001	0.001	0.001	0.001
	RMSE	0.033	0.028	0.032	0.032	0.033	0.029
	MAE	0.023	0.019	0.020	0.021	0.021	0.019

(LIME) have also been used in a few wind engineering studies to explain the machine learning predictions [19]. Therefore, the present study used both LIME and SHAP to explain the main features of the best-performing model.

3.3. Performance evaluation

To evaluate the performance of the ABC-ANN, GA-ANN, PSO-ANN, ICA-ANN, ANN, and CNN models, several metrics were used as indicators of model accuracy such as coefficient of determination (R^2), root mean square error (RMSE), mean square error (MSE), and mean absolute error (MAE). These performance metrics were calculated using the formulas provided in Eq. 4–7 respectively.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2} \tag{5}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 \tag{6}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \tag{7}$$

Here, n denotes the number of instances, $\bar{y}_i$, y_i and $\hat{y}_i$, and represent the observed average, observed, and predicted values of the response variable, respectively.

3.4. Workflow of the study

Fig. 3 shows the methodological framework of the study. The wind tunnel dataset was normalised using the MinMax Scaler in Sci-Kit Learn [27]. Also, one-hot encoding was applied to the variable "Surface". One-hot encoding assigns numerical digits to the feature value, which can convert a categorical variable to a numerical value to be used in ML. The dataset was divided into an 80:20 split for training and testing. Additionally, the training set was used with 5-fold cross-validation to optimise each model. Cross-validation ensures that the model is accurate and unbiased towards the training data. Once the model performance is evaluated, the best model is explained using SHAP and LIME to

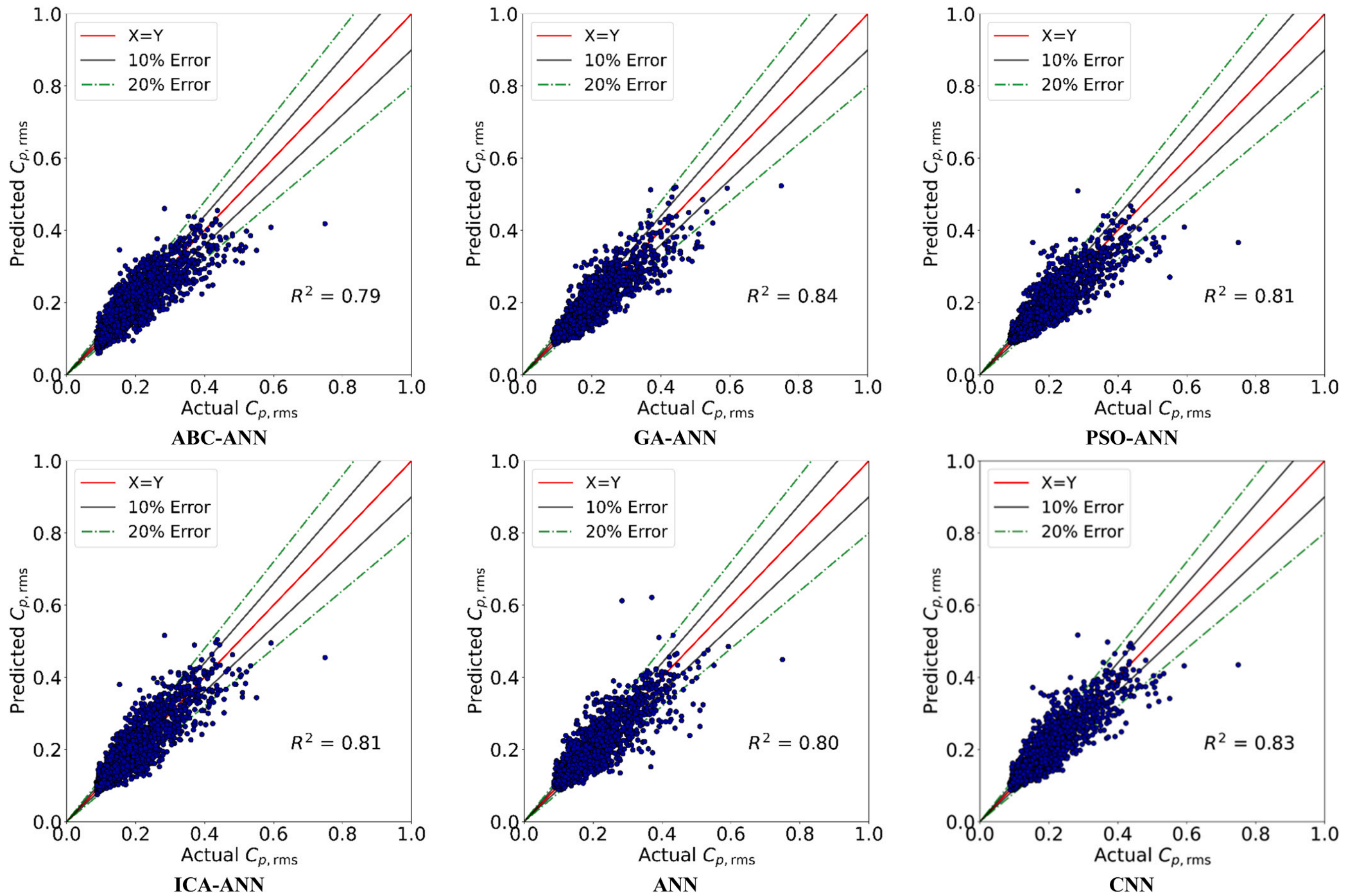


Fig. 5. Comparison of $C_{p,rms}$ predictions with the testing (wind tunnel data) $C_{p,rms}$.

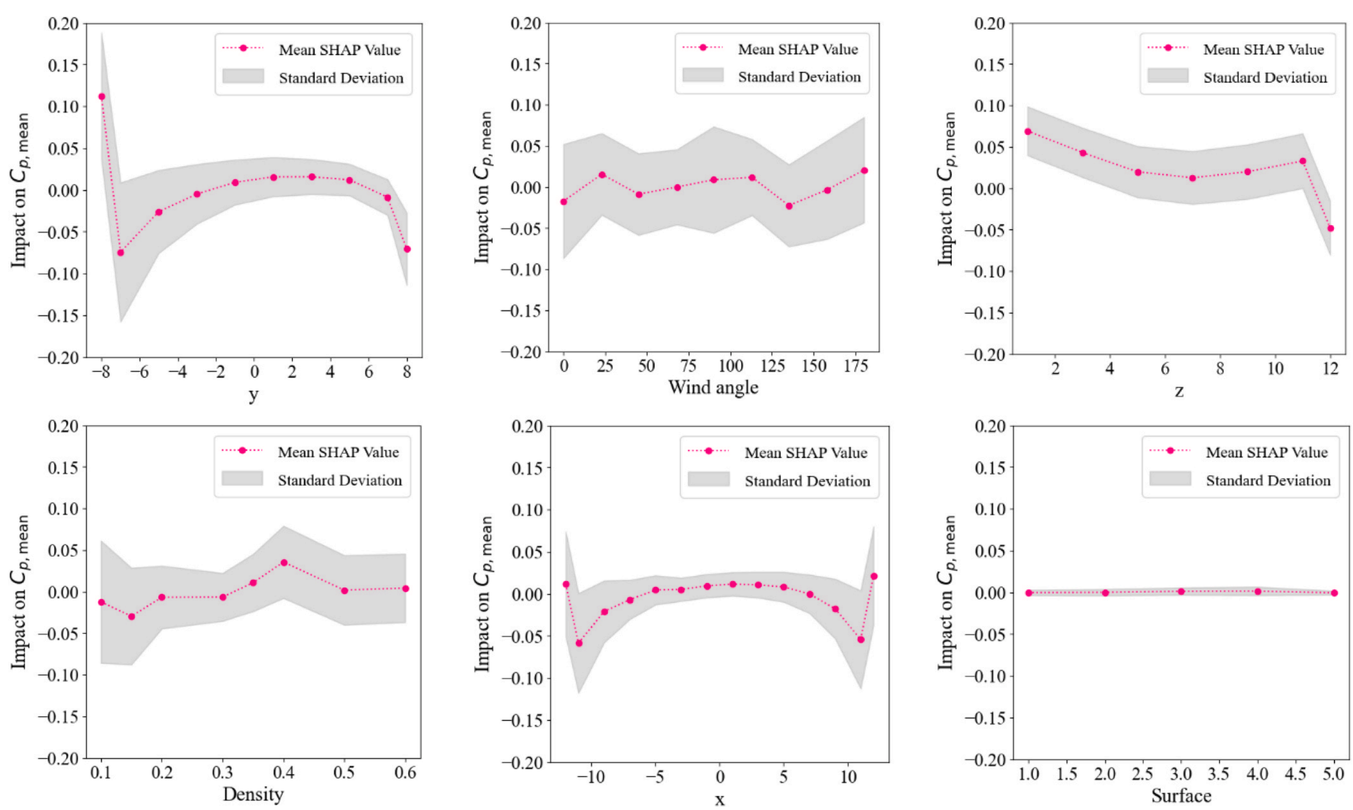


Fig. 6. Variation of SHAP value of each feature for $C_{p,mean}$ predictions (ranked in descending order of feature importance).

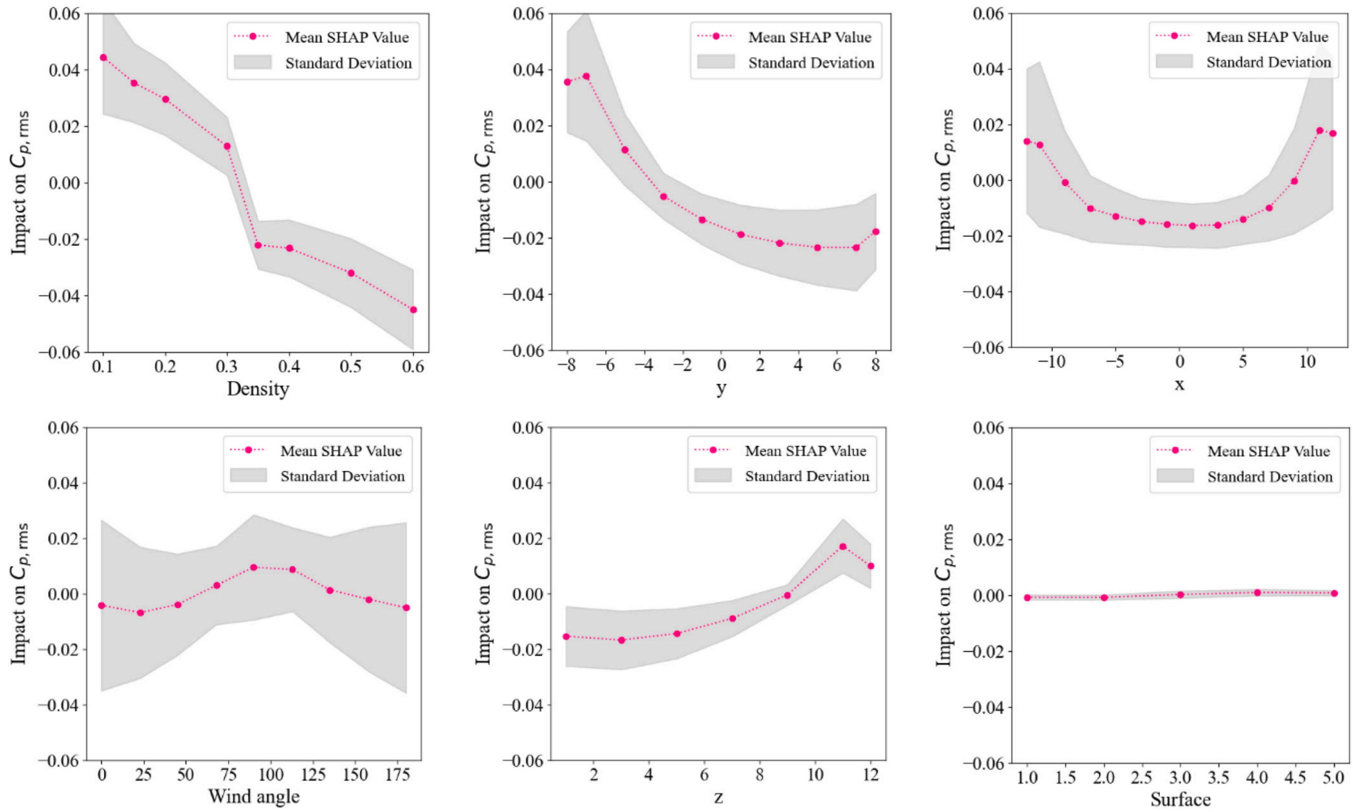


Fig. 7. Variation of SHAP value of each feature for $C_{p,rms}$ predictions (ranked in descending order of feature importance).

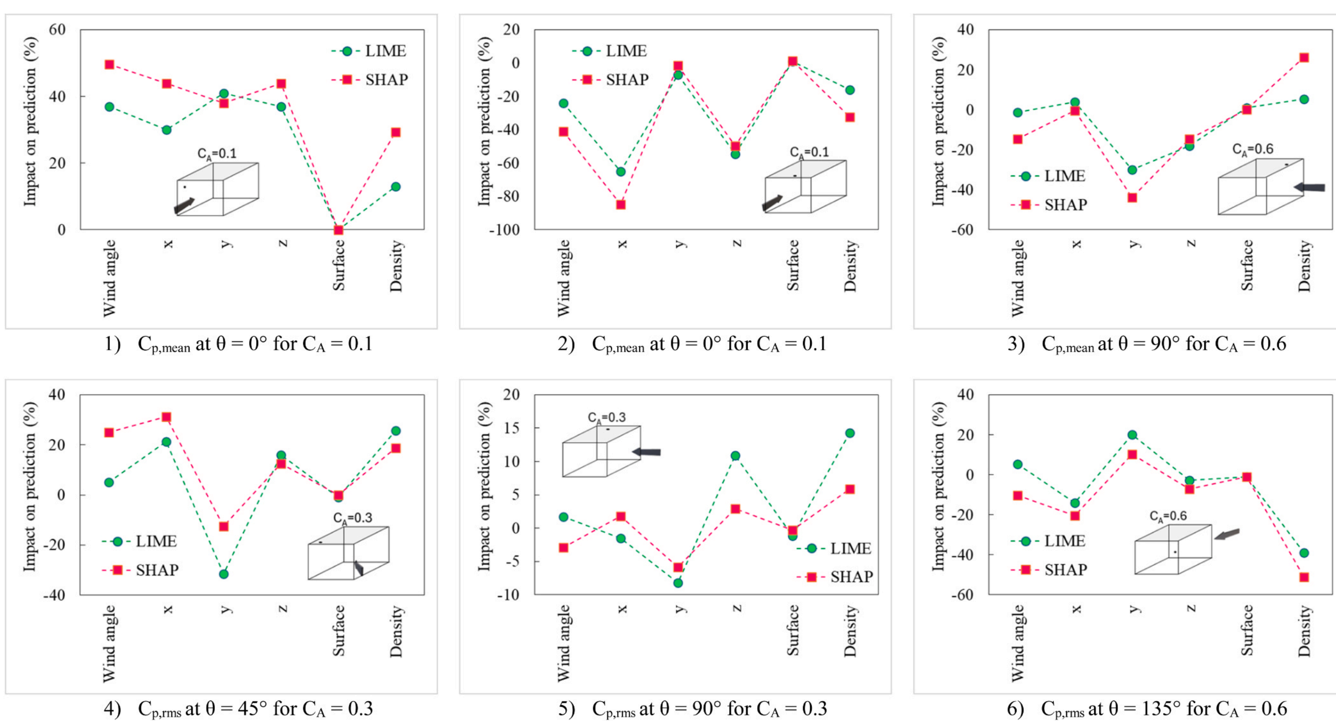


Fig. 8. Comparison of feature importance obtained from LIME and SHAP for selected instances.

investigate the importance of input features on the respective predictions. This is followed by a comparison of the model-predicted mean and fluctuating pressure with the results obtained from related work. The best model predictions are then compared from a wind engineering viewpoint to showcase how the model predicts crucial flow features and intense negative pressure at critical locations.

4. Results and discussion

This study developed hybrid neural networks to predict $C_{p,mean}$ and

Table 8
Comparison of prediction accuracies with related work.

Ref:	Building type	Prediction	Model	Training R^2	Testing R^2
[40]	Low rise building	$C_{p,mean}$	Gradient boosting tree	0.99	0.9
		$C_{p,rms}$		0.99	0.91
		$C_{p,max}$		0.98	0.83
		$C_{p,min}$		0.99	0.865
[15]	High rise building	$C_{p,mean}$	Decision tree	-	0.961
			Random forest	-	0.975
			Extreme gradient boost	-	0.976
			Generative adversarial network	-	0.988
		$C_{p,rms}$	Decision tree	-	0.837
			Random forest	-	0.888
			Extreme gradient boost	-	0.89
			Generative adversarial network	-	0.924
[41]	High rise building	$C_{p,mean}$	Support vector regression	0.92–0.95	-
			Random forest	0.92–0.95	-
			Gradient boosting tree	0.93–0.95	-
			Artificial neural network	0.94–0.96	-
		$C_{p,rms}$	Support vector regression	(−0.22)–0.4	-
			Random forest	(−0.18)–0.4	-
			Gradient boosting tree	(−0.08)–0.44	-
			Artificial neural network	(0.07)–057	-
[8]	Low rise building	$C_{p,mean}$	Decision tree	0.98	0.88
		$C_{p,rms}$	Decision tree	0.98	0.89
		$C_{p,max}$	Decision tree	0.98	0.87
		$C_{p,min}$	Decision tree	0.97	0.81
		$C_{p,mean}$	Random forest	0.99	0.91
		$C_{p,rms}$	Random forest	0.98	0.93
		$C_{p,max}$	Random forest	0.98	0.91
		$C_{p,min}$	Random forest	0.96	0.85
		$C_{p,mean}$	Extreme gradient boost	0.99	0.94
		$C_{p,rms}$	Extreme gradient boost	0.99	0.94
		$C_{p,max}$	Extreme gradient boost	0.98	0.93
		$C_{p,min}$	Extreme gradient boost	0.95	0.88
Present Study	Low rise building	$C_{p,mean}$	GA-ANN	0.98	0.96
		$C_{p,rms}$	GA-ANN	0.87	0.84

** $C_{p,max}$: instantaneous maximum pressure coefficient, $C_{p,min}$: instantaneous minimum pressure coefficient

$C_{p,rms}$ using geometric features such as x, y, and z (coordinates of pressure taps), surface of the building, wind direction, and surrounding building density (C_A). Four hybrid machine learning models, along with an individual ANN and CNN model, were utilised for this purpose. Hence, all six models were developed separately to predict both $C_{p,mean}$ and $C_{p,rms}$. The following sections present the results obtained from the models and model explanations. First, the authors evaluated the model efficiency using the time each algorithm takes for optimisation, training (training with cross validation) and testing. Table 3 shows the time taken for each algorithm in each phase. It is noted that the ANN and CNN were optimised using a grid search, considering a similar number of neurons in each layer. As the number of layers, learning rate, and neurons are changed in the grid, and the models were trained using 5-Fold cross validation. Therefore, compared to hybrid optimisation methods, the grid search approach took significantly more time to optimise. Methods like ABC, GA and PSO generally perform optimisation in an evolved manner, systematically finding the best architecture within even finer search spaces compared to grid search methods. The optimised architecture and the algorithm parameters are presented in Tables 4 and 5. Specifically, the prediction time of all neural networks is significantly lower than the time typically required for computational fluid dynamics (CFD) simulations reported in recent literature. Time-efficient turbulence modelling methods, such as Reynolds-Averaged Navier–Stokes (RANS) modelling, have been reported to require approximately 5 core hours for a mesh with one million elements [37]. In contrast, once trained, the neural networks were able to predict the mean pressure within 20 s for each wind direction, whereas the RANS model must be repeated for each specific angle, if needed. However, future studies are necessary to further analyse and compare numerical modelling techniques with data-driven approaches.

4.1. Mean pressure predictions ($C_{p,mean}$)

Table 6 shows the model accuracies for $C_{p,mean}$ predictions. According to the performance indices, all models performed at a good level; however, the GA-ANN model outperformed the other neural networks for the $C_{p,mean}$ predictions. The coefficient of determination (R^2) values were 0.976, 0.963, and 0.96 for the training, validation and testing, respectively. In addition, the root mean square error (RMSE) was 0.031, the MSE was 0.001, and the MAE was 0.022 in the training stage; and RMSE was 0.031, the MSE was 0.001, and the MAE was 0.022 in the validation stage. In the testing stage, the RMSE was 0.041, the MSE was 0.002, and the MAE was 0.027. It was observed that the training, validation and testing scores for all the models were in a comparable range, which showed more generalised models. The use of cross validation ensured that the models did not have any biases for training data. Furthermore, Fig. 4 illustrates the comparison of all models' performances based on the actual $C_{p,mean}$ values, alongside their respective predictions during the testing stage.

The GA-ANN model demonstrated the best performance, with minimal deviations exceeding the 10 % and 20 % error lines. Specifically, the GA-ANN method accurately predicts peak negative pressure coefficients, demonstrating its ability to capture complex wind pressure characteristics in an urban environment.

It was interesting to observe that the GA-ANN method outperformed the state-of-the-art extreme gradient boosting model developed by Meddage et al. [8] for predicting the mean wind pressure on low-rise buildings. While their model achieved an R^2 of 0.99 for training and 0.94 for testing, it showed a tendency to overfit. The authors believe that the nexus tree structure used in their study caused the overfitting in the results. In contrast, the GA-ANN model is less likely to overfit, providing a more accurate and generalised prediction of mean wind pressure on low-rise buildings in a complex urban environment. However, this does not rule out the applicability of the other methods, as they also achieved good accuracy.

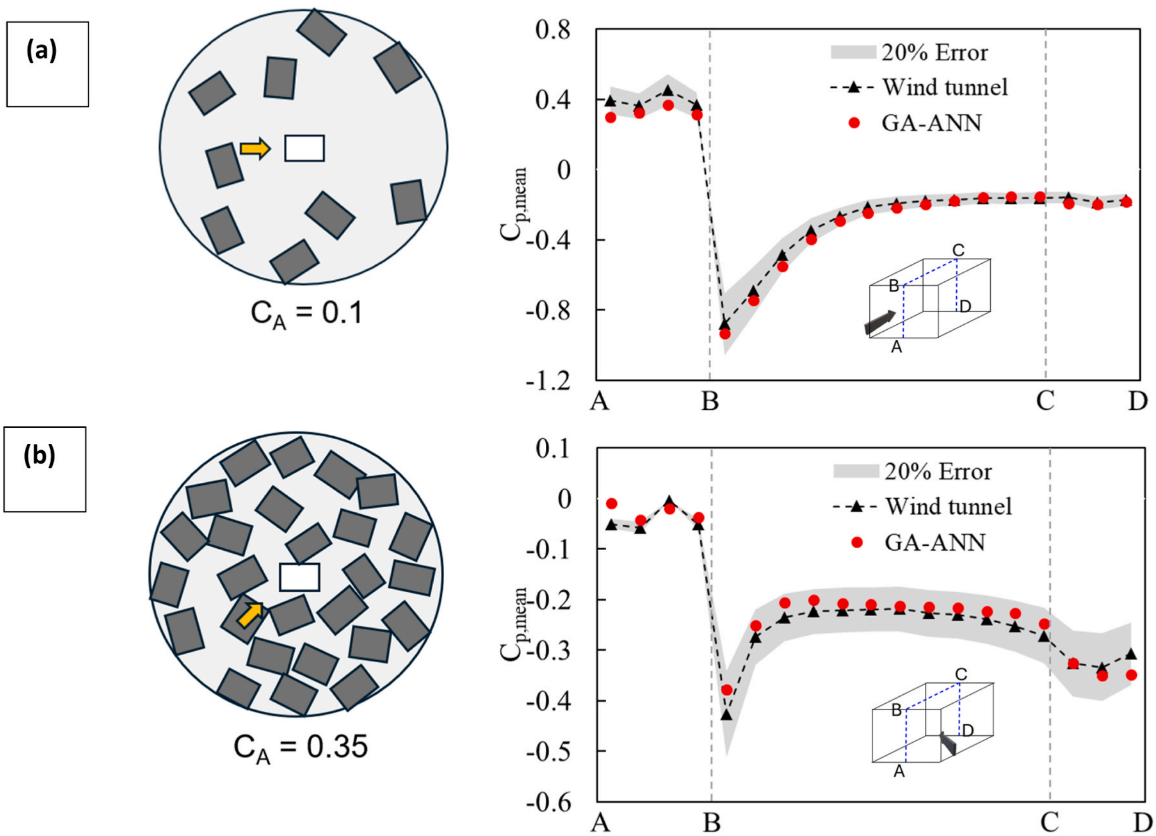


Fig. 9. $C_{p,mean}$ variation along the centerline of the building (a) $\theta = 0^\circ$ for $C_A = 0.1$, (b) $\theta = 45^\circ$ for $C_A = 0.35$ (θ: wind direction, C_A: density of surrounding buildings).

4.2. *Fluctuation pressure predictions ($C_{p,rms}$)*

Table 7 shows the model accuracies for fluctuation pressure coefficient ($C_{p,rms}$) predictions. The PSO-ANN, CNN and GA-ANN methods have predicted the fluctuation pressure coefficients with good accuracy. For higher fluctuations, the models tend to underpredict the $C_{p,rms}$ values, causing these deviations in accuracy. From a wind engineering perspective, GA-ANN and CNN predictions are acceptable and promising. It demonstrates the GA-ANN method's ability to accurately identify mean wind pressure and fluctuating wind pressures on low-rise buildings surrounded by geometrically similar buildings in a dense environment. The lowest prediction accuracy was observed for the ABC-ANN method for testing dataset (Fig. 5).

4.3. *Impact of geometric features, wind direction, and density on the predictions*

As the best model in both cases was the GA-ANN, the authors used SHAP and LIME to interpret the causality of the predictions. This is an important aspect of applied machine learning research to verify the consistency of the predictions using a theoretical background. Additionally, it improves end users' and domain experts' trust by providing the rationale behind the predictions.

The SHAP explanations for the $C_{p,mean}$ predictions are shown in Fig. 6 in descending order of feature importance. The geometric variable 'y' has the highest importance for the $C_{p,mean}$ predictions. Specifically, at y closer to −8, the SHAP value showcases a negative contribution, implying a steep decrease in $C_{p,mean}$ values. This can be attributed to the intense suction occurring near the leading corner of the roof ($y = -8$, $x = -10$, and $z = 12$) [38]. The SHAP values at each of these feature values show a steep drop in the predictions, indicating that the geometric parameters of the building govern the wind pressure on a low-rise building in a complex urban setting.

The second most dominant feature was the wind angle, where a negative SHAP value is observed at oblique wind directions. This confirms findings in wind engineering, which state that oblique wind directions (from θ=30° to 60° and from θ=120° to 150°) can exert intense negative pressure [38,39]. Both x and y coordinates indicate that at the edges of the building, a negative SHAP value (lower or negative $C_{p,mean}$) is anticipated. This is aerodynamically accurate, as the flow separation starts from the sharp edges of the building, creating steep pressure gradients and negative pressure [2]. The density of surrounding buildings is ranked fourth by SHAP, showing a constant increase in $C_{p,mean}$ with an increase in density of surrounding buildings (C_A).

In the SHAP explanations for $C_{p,rms}$ predictions, the density of surrounding buildings emerged as the most dominant feature. It indicates that at surrounding building density (C_A) from 0.1 to 0.2, higher fluctuations in wind pressure are observed on the target building. When the density increases further, the fluctuations decrease due to the dominance of the shielding effect. Nonetheless, oblique wind directions can still create higher fluctuations in wind pressure, as shown in Fig. 7. Similar to the SHAP explanations for mean pressure, the fluctuations are more likely to occur at the edges of the buildings. This can be a combined effect of building-induced turbulence and turbulence generated in the upstream (approaching wind).

Fig. 8 depicts the SHAP and LIME feature importances obtained for selected instances from the dataset. Both feature importances were normalised by the absolute sum of the feature importances of all features in that particular instance and presented as percentage values. SHAP and LIME show comparable feature importance values despite minor deviations in magnitude. The first instance, $C_{p,mean}$ at $\theta = 0^\circ$ for $C_A = 0.1$, shows positive mean pressure on the windward wall due to direct wind obstruction at $\theta = 0^\circ$ and flow stagnation. SHAP has assigned major importance to the wind angle accordingly whereas LIME has assigned

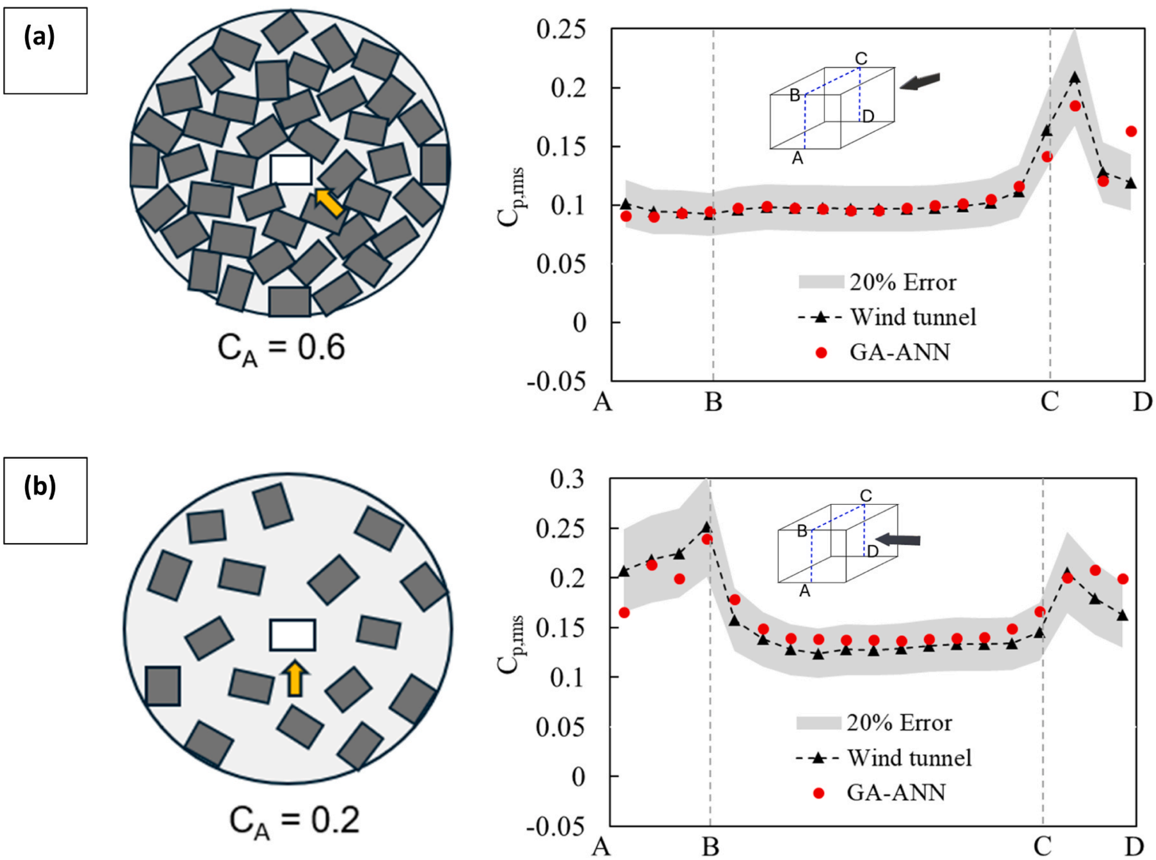


Fig. 10. $C_{p,rms}$ variation along the centerline of the building (a) $\theta = 135^\circ$ for $C_A = 0.6$, (b) $\theta = 90^\circ$ for $C_A = 0.2$ (θ: wind direction, C_A: density of surrounding buildings).

the majority to the y value.

The second and third instances illustrate the mean pressure on the roof, where flow separation occurs near the pressure tap considered. In each case, negative pressure is mainly caused by geometric parameters x and y, which denote the edges of the roof and create intense negative pressures due to flow separation. However, compared to the second instance, the C_A of the third instance has increased to 0.6, which reduces the magnitude of negative pressure, as indicated by the positive SHAP and LIME contributions. Similarly, SHAP and LIME importances for fluctuation pressure (Instances 4–6) show that increasing surrounding building density decreases the magnitude of positive fluctuation wind pressure on the roof, confirmed by the negative contribution in Instance 6. In Instance 4, the variable x has a positive impact on fluctuation pressure because the pressure tap is located on the edge of the roof where building-induced turbulence is significantly high, thereby increasing fluctuation pressure. Therefore, both SHAP and LIME explanations highlight that GA-ANN predictions adhere to principles generally accepted in wind engineering related to low-rise buildings. Moreover, SHAP and LIME are post hoc explanation methods which are applied after the models are trained, validated and tested. That is one advantage of these methods (SHAP, LIME) which do not interfere the accuracy of the models rather improve the interpretability of the predictions.

4.4. Comparison of GA-ANN predictions with the related work

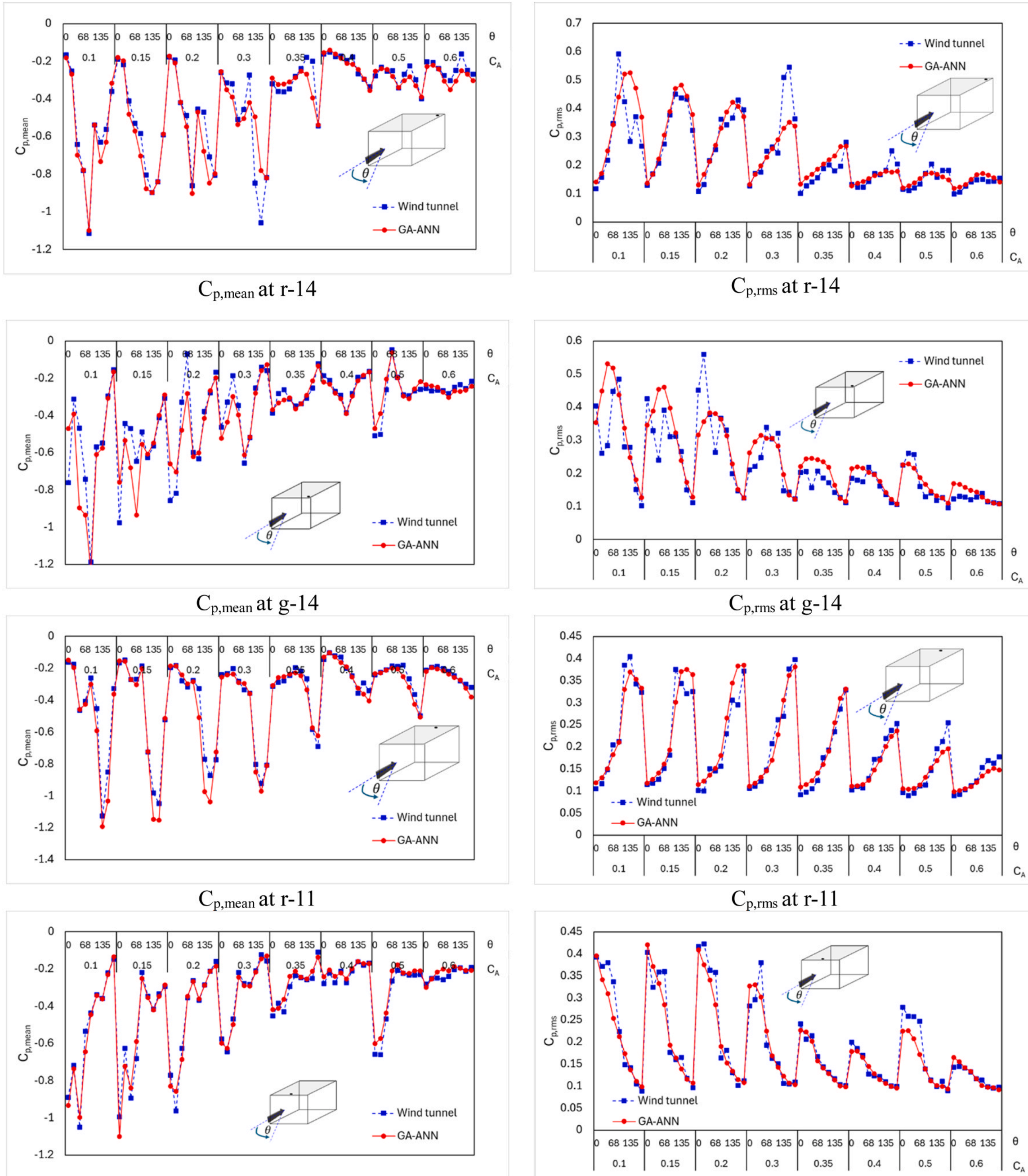
Following the model explanations, the authors reviewed the literature for similar studies that used classical machine learning approaches to predict wind pressure on buildings. Table 8 shows the prediction accuracies obtained from these studies. A common observation from previous work is the tendency towards overfitting, where the training accuracy is significantly higher than the testing accuracy. Despite these issues, tree-based methods have achieved reasonable accuracy for predictions. Compared to previous studies, the GA-ANN model significantly improved the accuracy of $C_{p,mean}$ predictions. While the predictive accuracy of $C_{p,rms}$ is lower compared to related work, the GA-ANN method overcomes the overfitting observed in other models. This confirms the ability of the GA-ANN method to accurately predict the wind pressure characteristics of low-rise buildings in the presence of surrounding buildings.

The overfitting observed in tree-based models is likely due to deviations in the wind tunnel data, even though the GA-ANN model is less affected by such deviations. A recent study by Glumac et al. [41] highlighted that individual ANN models may not effectively capture hidden patterns in wind pressure data, as evidenced by their lower R^2 values in model training. Therefore, hybrid models have an added advantage in predicting wind pressure on buildings, especially in complex urban configurations

4.5. Investigating the mean pressure and fluctuating pressure profiles on the building

After confirming the predictive performance of the genetic algorithm-based artificial neural network (GA-ANN), the mean ($C_{p,mean}$) and fluctuation ($C_{p,rms}$) wind pressure coefficient predictions along the centerline of the building were investigated. This analysis demonstrates how the GA-ANN model captures pressure gradients and flow separations compared to wind tunnel data.

Fig. 9.a shows the $C_{p,mean}$ variation along the centerline of the building at $\theta = 0^\circ$ for $C_A = 0.1$. Starting from point A, the GA-ANN model closely follows the wind tunnel data with very good accuracy. As the wind impacts the windward wall, positive pressure dominates the



$C_{p,mean}$ at r-14 $C_{p,rms}$ at r-14

$C_{p,mean}$ at g-14 $C_{p,rms}$ at g-14

$C_{p,mean}$ at r-11 $C_{p,rms}$ at r-11

$C_{p,mean}$ at g-11 $C_{p,rms}$ at g-11

Fig. 11. Directional variation of wind pressure coefficients at r-14, g-14, r-11, g-11 pressure taps (refer to Fig. 1), θ: wind direction, C_A: density of surrounding buildings.

entire surface. However, the density of surrounding buildings causes some shielding, resulting in a positive pressure of around 0.4. Typically, for isolated buildings, positive pressure around 0.8–0.9 can be observed on the windward wall. At point B, the flow starts separating, exerting intense negative pressure near the leading edge of the roof, which then decays over the trailing surface. The GA-ANN predictions accurately follow these variations, showing the model's ability to recognise special flow features. On the leeward wall, a mild uniform negative pressure is observed, and the GA-ANN model provides similar values.

Fig. 9.b shows the variation of mean pressure coefficient ($C_{p,mean}$) along the centerline of the building at wind direction (θ) = 45° for density of surrounding buildings (C_A) = 0.35. Along the AB line, small positive pressure coefficients are observed, with minor deviations in the predictions made by the GA-ANN model. Due to increased shielding from the surrounding buildings compared to the previous case, the magnitude of the negative pressure on the roof has decreased slightly. Additionally, the change in wind direction has contributed to this change in magnitude. Overall, the GA-ANN model accurately captured the behavior of wind pressure.

Fig. 10.a and b show the variation of the fluctuation pressure coefficient ($C_{p,rms}$) at different wind directions (θ = 135° and θ = 90°) and different surrounding building densities (0.6 and 0.2). Accordingly, the edges of the roof can experience higher fluctuations in wind pressure. Generally, where higher $C_{p,rms}$ is observed, higher peak wind pressure can be expected. Therefore, accurately predicting $C_{p,rms}$ demonstrates the GA-ANN model's ability to identify zones with higher peak pressure.

Although minor variations are observed in the predictions for certain instances, the overall deviations are within a 20 % error margin

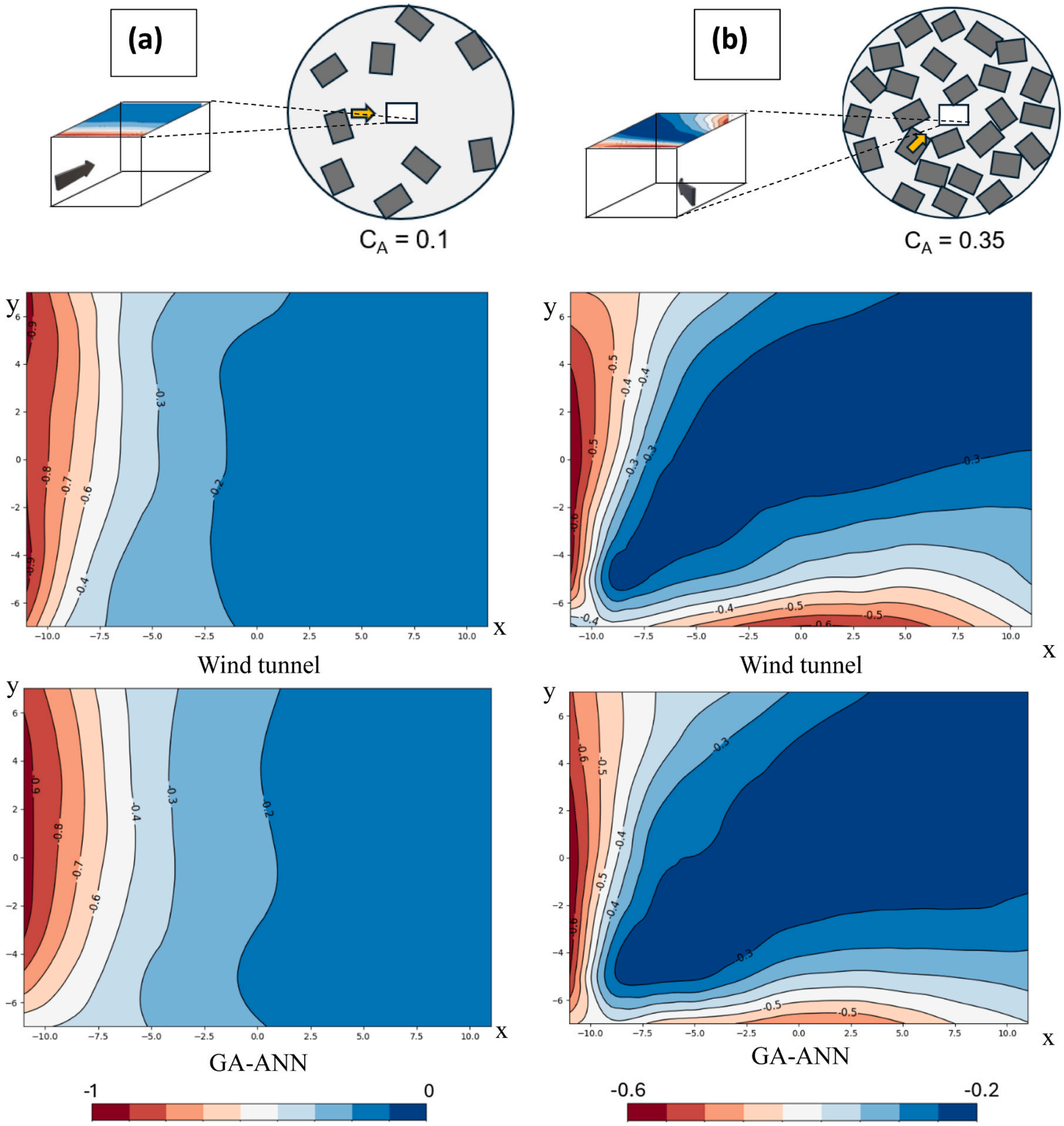


Fig. 12. Contours of $C_{p,mean}$ obtained from GA-ANN and wind tunnel data (a) $\theta = 0^\circ$ for $C_A = 0.1$, (b) $\theta = 45^\circ$ for $C_A = 0.35$ (θ: wind direction, C_A: density of surrounding buildings).

compared to the wind tunnel experimental data. Thus, the GA-ANN model can identify special flow features such as separation and zones with high turbulent fluctuations.

4.6. Comparison of directional pressure variation

The machine learning models were trained on wind tunnel data consisting of wind directions at $\theta = 0^\circ$ and $\theta = 180^\circ$ in an anticlockwise direction. At different wind angles, unique pressure distributions can occur on each surface. Oblique wind directions can create sudden changes in wind pressure and generate high suction, especially on the roof, due to intermittent flow separation and vortex formations. The authors chose pressure taps g-14, r-14, g-11, r-11 (refer to Fig. 1) to compare the wind tunnel results and GA-ANN predictions under different wind directions. As shown in Fig. 11, the GA-ANN method accurately captures the directional $C_{p,mean}$ and $C_{p,rms}$ variation all pressure taps. These taps are located at the corner of both leading edges of the roof, making it a crucial location that experiences intense negative pressure with high building-induced turbulent fluctuations.

At oblique wind directions, the GA-ANN model closely follows the high suction pressure observed in wind tunnel results. However, a few minor deviations were observed at densities of surrounding buildings (C_A) from 0.2 to 0.35 in the wind tunnel results and GA-ANN predictions which we believe were caused by the inconsistencies in the data. Regardless of those deviations, the GA-ANN model accurately reproduced the mean and fluctuating wind pressures at the locations on the roof where high building-induced turbulence is expected. Overall, as the surrounding building density increases, the GA-ANN tends to accurately predict the directional pressure variation of critical pressure taps.

4.7. Pressure contours obtained from GA-ANN with wind tunnel data

Pressure contours indicate how the predictions follow the spatial pressure distribution. They also reveal localised flow features such as pressure gradients, vortex formations, flow separation, and reattachment of the shear layer. The authors selected certain configurations for

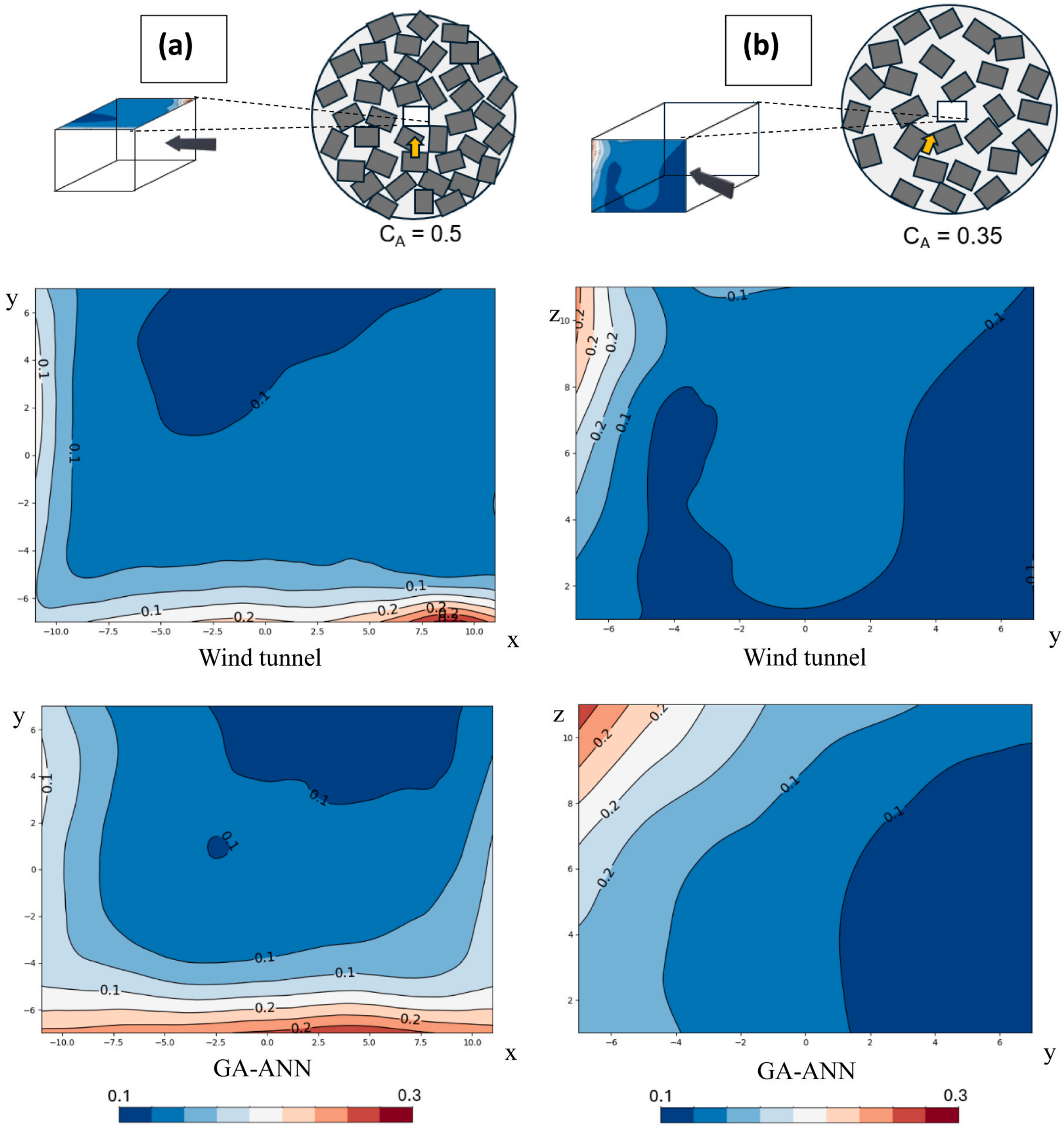


Fig. 13. Contours of $C_{p,rms}$ obtained from GA-ANN and wind tunnel data (a) $\theta = 90^\circ$ for $C_A = 0.5$, (b) $\theta = 68^\circ$ for $C_A = 0.35$ (θ: wind direction, C_A: density of surrounding buildings).

this analysis, as shown in Figs. 12 and 13, for $C_{p,mean}$ and $C_{p,rms}$, respectively. Fig. 12.a shows the variation of $C_{p,mean}$ over the roof surface at $C_A = 0.1$ for $\theta = 0^\circ$. The reason for selecting the roof surface is that state-of-the-art research in wind engineering suggests that roofs undergo intense peak wind loading and are more vulnerable to progressive collapse compared to the walls of low-rise buildings [39].

Starting from the leading edge of the roof, the wind flow begins to separate, creating intense negative pressure near the leading edge. However, over the trailing surface, it recovers to a certain extent and reaches -0.2. The genetic algorithm-artificial neural network (GA-ANN) predicted pressure contours show a good match with the wind tunnel pressure distribution. The pressure gradient near the leading edge is also comparable between the wind tunnel results and GA-ANN predictions. Fig. 12.b shows the same roof surface at $C_A = 0.35$ for $\theta = 45^\circ$. Generally, oblique wind directions create diagonally formed vortex bands from the leading corner of the roof. These regions can have intense negative pressure as a result of vortex formation.

Fig. 13 shows two instances of $C_{p,rms}$ predictions over the roof and the windward wall. Even though there were minor variations, the magnitudes of $C_{p,rms}$ are comparable between the wind tunnel results and GA-ANN predictions. In particular, the regions with high wind fluctuation pressures are reasonably modelled by the GA-ANN model. The GA-ANN model understands the effect of shielding (surrounding buildings) and the effect of different wind directions and closely matches the spatial mean and fluctuation pressure in each scenario. This is, therefore, a promising result in wind engineering studies on low-rise buildings. Compared to the machine learning predictions obtained from an isolated building, the present workflow seems pragmatic as it reflects the real-world conditions and the applicability of machine learning methods in wind engineering applications related to low-rise buildings. The GA-ANN method can be used as a complementary approach for predicting wind pressure on low-rise buildings in an urban setting.

5. Conclusions

This study used hybrid neural network methods (four hybrid neural networks, each combined with artificial bee colony (ABC), genetic algorithm (GA), particle swarm optimisation (PSO), and independent component analysis (ICA), along with an individual artificial neural network (ANN) model and a convolutional neural network (CNN) to predict the wind pressure on low rise buildings in a dense urban environment. The following are the remarks of our study

- Hybrid neural networks and the individual artificial neural network (ANN) model accurately predicted the mean and fluctuating wind pressures on the low-rise building. The coefficient of determination (R^2) of the training predictions ranged from 0.884 to 0.976 and the testing predictions ranged from 0.846 to 0.96 for mean pressure predictions. The R^2 of the training predictions ranged from 0.825 to 0.877 and the testing predictions ranged from 0.790 to 0.84 for fluctuating pressure predictions.
- The genetic algorithm-artificial neural network (GA-ANN) method looked promising for predicting mean and fluctuating pressures with the highest accuracy compared to the remaining methods. It achieved R^2 of 0.96 for mean pressure prediction and R^2 of 0.84 for fluctuation pressure predictions in the test set, implying a good generalisation ability. The root mean square error (RMSE) of GA-ANN predictions was 0.041 and 0.028 for mean and fluctuation pressure predictions in the test set, respectively. In addition, the CNN predictions were also comparable with GA-ANN for fluctuating pressure predictions.
- Shapley additive explanations (SHAP) and Local interpretable model agnostic explanations (LIME) on the GA-ANN predictions showed that geometric features govern the mean wind pressure, while the surrounding building density governs the fluctuating wind pressure on low-rise buildings in a complex urban configuration. These explanations adhere to general principles in wind engineering, confirming the GA-ANN model's understanding of the complex nature of wind pressure.
- In comparison with the related work, state of the art machine learning models used in wind engineering application showcased relatively lower accuracy and tendencies of overfitting. Interestingly, the GA-ANN has improved the accuracy of wind pressure predictions and obtained more generalised models without overfitting.
- The GA-ANN predicted wind pressure closely followed the wind tunnel results and modelled the wind pressure on each surface to a reasonably good level. Localised flow features such as flow separation and intermittent vortex formation have been well captured by the GA-ANN method, implying the ability of the model to distinguish the critical regions on the building with intense negative pressure.
- Moreover, the GA-ANN model accurately reconstructed the directional variation of the pressure and shielding effect on the pressure on the target building by surrounding buildings. It shows the ability of hybrid neural network approaches to accurately model the wind pressure on low-rise buildings in a real urban environment.

CRediT authorship contribution statement

Upaka Rathnayake: Resources, Writing – review & editing. **Meddage D. P. P.:** Writing – review & editing, Conceptualization, Supervision. **Hasanthi Wijesundara:** Data curation, Methodology, Investigation. **Sumudu Herath:** Supervision, Writing – review & editing. **Hirushan Sajindra:** Visualization, Data curation, Software, Writing – original draft, Formal analysis. **Shashika Dharmawansha:** Methodology, Visualization, Validation.

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.

References

[1] Meddage D, Lewangamage C, Weerasuriya AU. On the deviation of mean pressure coefficients in wind loading standards for a low-rise, gable-roofed building with boundary walls. Structures, 36. Elsevier; 2022. p. 50–64.
[2] Holmes JD. Wind pressures on tropical housing. J Wind Eng Ind Aerodyn 1994;53 (1-2):105–23.
[3] Holmes J.D. Wind loads on low rise buildings: a review; 1983.
[4] Davenport A. Wind loads on low-rise buildings: final report of phases I and II. Eng Sci Rep BLWTSS8 1977.
[5] Stathopoulos T. Wind loads on eaves of low buildings. J Struct Div 1981;107(10): 1921–34.
[6] Stathopoulos T, Luchian H. Wind pressures on buildings with stepped roofs. Can J Civ Eng 1990;17(4):569–77.
[7] Stathopoulos T, Saathoff P. Wind pressure on roofs of various geometries. J Wind Eng Ind Aerodyn 1991;38(2-3):273–84.
[8] Meddage D, et al. Explainable Machine Learning (XML) to predict external wind pressure of a low-rise building in urban-like settings. J Wind Eng Ind Aerodyn 2022;226:105027.
[9] Blocken B. 50 years of computational wind engineering: past, present and future. J Wind Eng Ind Aerodyn 2014;129:69–102.
[10] Quang TV, Doan DT, Phuong NL, Yun GY. Data-driven prediction of indoor airflow distribution in naturally ventilated residential buildings using combined CFD simulation and machine learning (ML) approach. J Build Phys 2024;47(4):439–71.
[11] Chen Y, Kopp G, Surry D. Interpolation of wind-induced pressure time series with an artificial neural network. J Wind Eng Ind Aerodyn 2002;90(6):589–615.
[12] Chen Y, Kopp G, Surry D. Prediction of pressure coefficients on roofs of low buildings using artificial neural networks. J Wind Eng Ind Aerodyn 2003;91(3): 423–41.
[13] Vrachimi I, Paula Melo A, Cóstola D. Prediction of wind pressure coefficients in building energy simulation using machine learning. Building Simulation 2017, 15. IBPSA; 2017. p. 2467–74.
[14] Ma X, Xu F, Chen B. Interpolation of wind pressures using Gaussian process regression. J Wind Eng Ind Aerodyn 2019;188:30–42.
[15] Hu G, Liu L, Tao D, Song J, Tse KT, Kwok KC. Deep learning-based investigation of wind pressures on tall building under interference effects. J Wind Eng Ind Aerodyn 2020;201:104138.
[16] Huang Y, Ou G, Fu J, Zhang H. Prediction of mean and RMS wind pressure coefficients for low-rise buildings using deep neural networks. Eng Struct 2023; 274:115149.
[17] Rajaee T, Khani S, Ravansalar M. Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: a review. Chemom Intell Lab Syst 2020;200:103978.
[18] Tao H, et al. Hybridized artificial intelligence models with nature-inspired algorithms for river flow modeling: a comprehensive review, assessment, and possible future research directions. Eng Appl Artif Intell 2024;129:107559.
[19] Meddage D, Ekanayake IU, Weerasuriya A, Lewangamage C. Tree-based regression models for predicting external wind pressure of a building with an unconventional configuration. 2021 Moratuwa Engineering Research Conference (MERCon). IEEE; 2021. p. 257–62.
[20] Yan B, et al. Explainable machine learning-based prediction for aerodynamic interference of a low-rise building on a high-rise building. J Build Eng 2024;82: 108285.
[21] Tamura Y., Aerodynamic database for low-rise buildings, Global Center of Excellence Program, Tokyo Polytechnic Univ., Tokyo, Japan, ⟨http://www⟩.wind. arch. t-kougei. ac. jp/system/contents/code/tpu⟩(Dec. 12, 2013); 2012.
[22] King M-F, et al. Investigating the influence of neighbouring structures on natural ventilation potential of a full-scale cubical building using time-dependent CFD. J Wind Eng Ind Aerodyn 2017;169:265–79.
[23] Elshaer A, Gairola A, Adamek K, Bitsuamlak G. Variations in wind load on tall buildings due to urban development. Sustain Cities Soc 2017;34:264–77 (ed).
[24] Aristodemou E, et al. How tall buildings affect turbulent air flows and dispersion of pollution within a neighbourhood. Environ Pollut 2018;233:782–96.
[25] AIJ-RLB. Recommendations for Loads on Buildings. Japan Tokyo: Architecture Institute of; 2004.
[26] Shukla N, Fricklas K. Machine Learning with TensorFlow. Greenwich: Manning; 2018.
[27] Pedregosa F, et al. Scikit-learn: machine learning in python. J Mach Learn Res 2011;12:2825–30.
[28] Kaya E, Gorkemli B, Akay B, Karaboga D. A review on the studies employing artificial bee colony algorithm to solve combinatorial optimization problems. Eng Appl Artif Intell 2022;115:105311.
[29] Hancer E, Xue B, Zhang M, Karaboga D, Akay B. Pareto front feature selection based on artificial bee colony optimization. Inf Sci 2018;422:462–79.
[30] Le LT, Nguyen H, Dou J, Zhou J. A comparative study of PSO-ANN, GA-ANN, ICA-ANN, and ABC-ANN in estimating the heating load of buildings' energy efficiency for smart city planning. Appl Sci 2019;9(13):2630.
[31] Khandelwal M, Armaghani DJ. Prediction of drillability of rocks with strength properties using a hybrid GA-ANN technique. Geotech Geol Eng 2016;34(2): 605–20.

[32] Mohamad ET, Jahed Armaghani D, Momeni E, Alavi Nezhad Khalil Abad SV. Prediction of the unconfined compressive strength of soft rocks: a PSO-based ANN approach. Bull Eng Geol Environ 2015;74:745–57.

[33] Karhunen J, Oja E, Wang L, Vigario R, Joutsensalo J. A class of neural networks for independent component analysis. IEEE Trans Neural Netw 1997;8(3):486–504.

[34] Lundberg SM, Lee S-I. A unified approach to interpreting model predictions. Adv Neural Inf Process Syst 2017;30.

[35] Thisovithan P, Aththanayake H, Meddage D, Ekanayake I, Rathnayake U. A novel explainable AI-based approach to estimate the natural period of vibration of masonry infill reinforced concrete frame structures using different machine learning techniques. Results Eng 2023;19:101388.

[36] Ekanayake I, Meddage D, Rathnayake U. A novel approach to explain the black-box nature of machine learning in compressive strength predictions of concrete using Shapley additive explanations (SHAP). Case Stud Constr Mater 2022;16:e01059.

[37] Bre F, Gimenez JM. A Cloud-based Platform to Predict Wind Pressure Coefficients on Buildings. Building simulation, 15. Springer; 2022. p. 1507–25.

[38] Holmes JD, Paton C, Kerwin R. Wind Loading of Structures. CRC press; 2007.

[39] Parackal KI, Ginger JD, Henderson DJ. Progressive failures of batten to rafter connections under fluctuating wind loads. Eng Struct 2020;215:110684.

[40] Weng Y, Paal SG. Machine learning-based wind pressure prediction of low-rise non-isolated buildings. Eng Struct 2022;258:114148.

[41] Glumac AŠ, Jadhav O, Despotović V, Blocken B, Bordas SP. A multi-fidelity wind surface pressure assessment via machine learning: a high-rise building case. Build Environ 2023;234:110135.