



Accelerated environmental performance-driven urban design with generative adversarial network



Chenyu Huang^a, Gengjia Zhang^b, Jiawei Yao^{c,d,*}, Xiaoxin Wang^e, John Kaiser Calautit^f, Cairong Zhao^g, Na An^c, Xi Peng^h

^a School of Architecture and Art, North China University of Technology, Beijing, 100144, China

^b Department of Architecture, Tamkang University, 25137, Taiwan, China

^c College of Architecture and Urban Planning, Tongji University, Shanghai, 200092, China

^d Key Laboratory of Ecology and Energy-saving Study of Dense Habitat (Tongji University), Ministry of Education, Shanghai, 200092, China

^e Faculty of Urban Architecture, Wuhan City College, Wuhan, 430075, China

^f Department of Architecture and Built Environment, University of Nottingham, Nottingham, NG7 2RD, UK

^g Department of Computer Science and Technology, Tongji University, Shanghai, 201804, China

^h Faculty of Architecture and Urban Planning, University of Stuttgart, Keplerstraße 11, 70174, Stuttgart, Germany

ARTICLE INFO

Keywords:

Performance-driven design
Generative adversarial network
Surrogate model
Genetic optimization
Urban block

ABSTRACT

The morphological design of urban blocks greatly affects the outdoor environment. Currently, performance-based urban and building design relies on a time-consuming numerical simulation process, hindering performance optimization early in the design process. This paper proposes an automated design process that applies generative adversarial network (GAN) as a surrogate model to accelerate environmental performance-driven urban design. Parameterized urban blocks are designed for random sampling and constructing a numerical simulation database. The GAN model was trained to predict pedestrian level wind (PLW), annual cumulative solar radiation (Radiation) and Universal Thermal Climate Index (UTCI) in real-time. The GAN-based surrogate model is combined with a multi-objective genetic algorithm to achieve real-time optimization of urban morphology. The results show that on the test set, the pix2pix model using a specific encoding method predicts the R^2 of 0.70, 0.86 and 0.80 for PLW, Radiation and UTCI, respectively, while the method can speed up 120–240 times compared to the numerical simulation method. The optimization results show that NSGA-II combined with global averaging pooling achieves the best optimization results. When the number of optimized samples exceeds 174, the proposed method has a time advantage over traditional methods for outdoor environment optimization in urban design.

1. Introduction

1.1. Background

By 2050, the global population will exceed 10 billion, with 68% residing in urban areas [1]. The heat island effect illustrates how metropolitan environments influenced by human activity have ventilation and radiation characteristics distinct from the natural environment. In cities where temperatures are increasing, and wind speeds are decreasing, populations are exposed to severe urban environments and have to confront health and safety concerns that have already resulted in numerous deaths and injuries [2,3]. Urban buildings with a high density diminish public space and are detrimental to urban ventilation and

pollutant dispersion. In 2003, the SARS virus accumulated in an area of Hong Kong with poor ventilation and infected pedestrians [4]. Today, COVID-19 has become the new normal, and designers must pay closer attention to the consequences of urban environment on urban populations [5].

There is a close relationship between urban form and urban environment. Macrolevel low-carbon urban planning, mesolevel sustainable urban design, and microlevel green building design all contribute to establishing a sustainable future urban environment. The decarbonization and energy efficiency goals of economies are now the driving factor in developing high-performance buildings, and many energy and environmental evaluation standards are being enforced, even though the cost of achieving them is high. In China, all new buildings are mandated

* Corresponding author. College of Architecture and Urban Planning, Tongji University, Shanghai, 200092, China.

E-mail address: jiawei.yao@tongji.edu.cn (J. Yao).

to strictly control energy performance and carbon emissions according to the latest standards (GB 55015–2021). It poses a challenge for building design. On the one hand, increased building performance leads to higher costs, and fortunately, many countries and regions offer economic subsidies for energy-efficient and environmental-friendly buildings. On the other hand, the importance of building performance evaluation and optimization is emphasized, and the design process has to use complex and time-consuming simulation tools to ensure that the design results pass performance reviews.

Traditionally, designers have focused on creating indoor spaces with natural ventilation and daylighting. With the development of numerical modelling software, there has been more research on establishing an environmental-friendly urban block. Prior studies have fully explored the impact of urban morphological indicators such as building height, building density, building volume [6], floor area ratio [7], frontal area ratio [8], and sky view factor [9] on urban environmental performance. Regarding outdoor environmental performance, pedestrian-level wind [10], outdoor thermal comfort [11], sunlight hours and solar radiation [12], photovoltaic power generation [13], energy intensity [14], and carbon emissions [15] are of significant relevance. These studies provide a solid foundation for enhancing the environmental performance of urban areas. However, it is difficult for the above findings to provide specific recommendations for the design process due to the wide variety of urban block design prototypes.

Numerous researchers have stressed design strategies to build a direct correlation between design proposals and environmental performance. The performance evaluation is often a final review before design completion in traditional design processes. During the process, a professional division of labour occurs, with architects emphasizing morphological design while HVAC and green building engineers assessing environmental performance [16]. This strategy separates the design process and performance optimization, commonly known as the “post-evaluation” paradigm [17]. This separation may lead to an incomplete optimization of building performance. It has been demonstrated that improving designs in later stages results in tiny benefits with huge costs [18]. Extensive research has been conducted towards the early stages of design [19].

On the one hand, to improve the interactivity of performance evaluation during the design process, some researchers have built data and visual interfaces between numerical simulation software and 3D modelling platforms [20,21]. Ladybug tools have been developed in multiple versions embedded in mainstream 3D modelling platforms and provide user interfaces to environmental performance simulation engines such as Radiance, EnergyPlus, and OpenFOAM. These software and plug-ins make it much easier to evaluate the environmental performance of buildings in the early design stages.

On the other hand, the time consumption of simulation engines hinders the performance optimization in the early design stages, and as a result, scholars have investigated techniques to accelerate environmental performance evaluation. Some scholars use simplified physical models to accelerate the computational process from the perspective of the underlying algorithms of performance simulation. The simulation of large-scale wind fields using finite difference methods of computational fluid dynamics (CFD) is time-intensive. The Reynolds Averaged Navier-Stokes (RANS) and Large Eddy Simulations (LES) methods have significantly reduced the time required relative to the Direct Numerical Simulation (DNS) method in modelling the airflow. In urban areas, evaluating different wind directions requires at least several hours and is often longer. Some researchers have employed fast fluid dynamics (FFD) to drastically reduce the simulation duration of wind environments. FFD is created for computer graphics and fluid rendering with a three-step advanced format to solve the Navier-Stokes equations and continuity equations for incompressible viscous fluids [18,22,23]. With the advancement of computer processing efficiency and technology, other scholars discuss using hardware acceleration, particularly GPU parallel computing, to speed up environmental performance simulations [24,

25].

However, the basic principles of the above methods are similar to the mainstream methods, the speed-up is limited, and real-time feedback is still challenging to obtain in design process. Some scholars have discussed data-driven performance evaluation with the rise of machine learning (ML) and deep learning (DL) techniques. This type of research packages white-box models based on performance simulations into black-box or grey-box models through learning and invokes pre-trained models for performance prediction with high efficiency and accuracy.

1.2. Our study

In order to efficiently predict the outdoor environment at the urban block scale and provide high-performance urban design solutions in a short time, we developed an efficient optimization method based on generative adversarial networks (GANs).

As a popular branch of DL, GANs were used to predict the outdoor environment with full information rather than characterized by a single value. We compared two typical GAN models, pix2pix and cycleGAN, for predicting urban pedestrian level wind (PLW) annual cumulative solar radiation (Radiation) and Universal Thermal Climate Index (UTCI). Since GAN is traditionally applied to learning natural images, such as animal photos and urban street scenes, one of our contributions is to discuss how to make the outdoor environmental data into a useable format for GANs and evaluate the performance of different encoding approaches. In addition, to demonstrate that the prediction results of the GAN-based surrogate model have fidelity, we compared it with a supervised regression ML model.

The dataset used for training was obtained from simulations, and the parameter settings of the simulations were validated using wind tunnel experiments. The full data was divided into a training set and a test set, where we trained on the training set and evaluated the performance of the model on the test set. We extracted the typical morphology of European urban blocks and parametrically redesigned nine urban blocks. The geometries were randomly sampled by varying design parameters to increase the generalizability of the GAN model to predict outdoor environments.

Many studies and practices use heuristic optimization algorithms in the early design stages. Although GANs as surrogate models can provide more detailed information about the outdoor environment, they also bring difficulties in accessing objective for optimization algorithms. Each optimization objective can only be a single value. We discuss three ways to reduce the dimensionality of the output of GAN, namely Standard Deviation (STD), Principal Components Analysis (PCA), and Global Average Pooling (GAP). The effects of different dimensionality reduction methods on the optimization results were compared.

This paper is organized as follows: Section 2 reviews the application of optimization algorithms in the early stages of design and related studies where ML and DL were used for environmental performance prediction; Section 3 explains the complete workflow of the study, including the methods of data preparation, the architecture of GAN, and the application of optimization algorithms; Section 4 describes the dataset, visualizes the prediction results of GAN, and gives the quantitative evaluation of the model performance. In addition, in Section 4, we discuss the effect of the encoding method on GAN prediction performance, the effect of the dimensionality reduction method on optimization results of the optimization algorithm and the difference in performance and optimization results between supervised regression-based and GAN-based surrogate models; finally, Section 5 summarizes the main conclusions of this paper.

2. Related works

2.1. Performance optimization in the early stages of design

The subjective trial-and-error-based design decisions are inefficient

due to architect knowledge background limitations [26]. Some researchers have included optimization algorithms in the environmental performance optimization process in the early design stages. With the development of parametric design, the architectural design process can be fully described by data processing, which has contributed to the prosperity of optimization algorithms in the design field. In the past, external optimization tools such as MATLAB and Isight are often used [27]. However, the lack of usability of such tools has discouraged many architects. Currently, evolutionary algorithm plug-ins such as genetic algorithms and simulated annealing algorithms have been integrated into mainstream parametric design platforms such as Rhino/Grasshopper and Revit/Dynamo. In conjunction with performance evaluation plug-ins, these tools have significantly improved environmental performance optimization in the early stages of design [28]. This method of design is known as performance-driven design and includes parametric design, performance evaluation, and optimization [29]. Multi-objective optimization is utilized to determine the Pareto non-inferior solution set of design proposals when there are many environmental indicators to be improved and mutually exclusive. Bahdad et al. (2021) optimized the shading components using a multi-objective genetic algorithm to achieve the optimal interior lighting and thermal environment Bahdad et al. (2021) [30]. Nasrollahzadeh (2021) optimized residential units' energy usage, internal daylight, and thermal comfort by modifying the building envelope using a multi-objective method Nasrollahzadeh (2021) [31]. At the urban block scale, Natanian and Wortmann (2021) performed a multi-objective optimization of residential building morphology to obtain maximum solar energy and sunlight hours Natanian and Wortmann (2021) [32]. Mirzabeigi and Razkenari (2022) proposed a multiphase optimization framework for conceptual urban design, establishing a baseline model that compares the energy performance of each alternative scenario with the outdoor environment (including the outdoor wind environment and thermal comfort) to a baseline scenario to determine the optimal one Mirzabeigi and Razkenari (2022) [33]. Chang et al. (2019) optimized the sky visibility factor, solar radiation, and energy performance of campus layouts using reinforcement learning Chang et al. (2019) [34].

However, inefficient performance simulations are a significant barrier to the performance-driven urban design. Specifically, when the scale of a geometry increases, the complexity of wind environment simulation, average radiation temperature computation, and urban energy modelling increases dramatically, necessitating a nonlinear increase in computer processing power and processing time [23,35]. It prevents optimization algorithms from receiving performance data on the present design solution on time, impeding the optimization process's progress. In the case of genetic algorithms, for instance, acquiring the Pareto optimal solution set frequently requires multiple generations of evolution and iteration. The main time consumption in applying genetic algorithms is acquiring the objective values. There is a paradox in that acquiring the optimal global solution of a genetic algorithm relies on a sufficient number of individuals to explore. However, the increased number of individuals increases the simulation time consumption of the objective value. Numerical simulation software allows a user to balance model complexity and duration. As a compromise, the designer must simplify the model and choose coarse simulation settings, such as a larger grid size, to minimize the simulation time to ensure that the optimization process operates well. It diminishes the reliability of the environmental performance evaluation results and the quality of the ideal solution. Lin et al. (2021) outlined significant difficulties that influence the implementation of performance-driven design processes in the early design stage, including model integration, real-time performance analysis, and interactive design optimization Lin et al. (2021) [36].

2.2. Application of ML in building performance optimization

In the past few years, data-driven approaches have borne fruit in

areas such as energy modelling [37], energy scheduling [38], and load forecasting [39]. Many scholars are now focusing on using machine learning to predict the environmental performance of design solutions in the early design stages. Supervised regression algorithms provide designers with a way to roughly estimate building performance in real-time with only the necessary inputs by learning the correspondence between the input parameters of a building and its environmental performance indicators. This approach gives designers a better understanding of what influences the environmental performance of a design solution and increases their control over the performance optimization process. Weerasuriya et al. (2020) combined artificial neural network (ANN)-based surrogate models, optimization algorithms (genetic algorithms), and CFD simulations to find the optimal lift-up design that maximizes pedestrians' wind comfort or thermal comfort Weerasuriya et al. (2020) [40]. Han et al. (2021) developed an ANN-based modelling approach for predicting annual daylight performance in the early design stages Han et al. (2021) [41]. S. Wang et al. (2021) optimized the visual comfort and outdoor thermal performance of high-rise residential layouts with an ANN applied to speed up the simulation process Wang et al. (2021) [42]. Wang et al. (2021) deployed a neural network-based energy prediction model in energy-driven urban design, defining three optimization objectives: maximum solar energy utilization, solar lighting of the first floor, and minimum building energy demand Wang et al. (2021) [43]. Wu et al. (2021) combined parametric residential building design with an evolutionary optimization algorithm based on a surrogate model in which machine learning is used to predict the wind environment to accelerate the convergence of the optimization algorithm Wu et al. (2021) [44]. Using classic machine learning regression techniques, such as multivariate linear regression (MLR), support vector regression (SVR), a multilayer perceptron (MLP), and random forest (RF), the above investigations are often geared at continuous environmental performance indicators of buildings. Faced with the environmental performance indicators with spatial distribution, such as pedestrian level wind, cumulative solar radiation, and UTCI, simply forecasting the average or eigenvalues will result in the loss of local detail information and incomplete estimation. According to turbulence theory, outdoor airflow spreads in space in a complicated manner after exchanging energy and momentum with buildings, which classical machine learning algorithms struggle to learn and predict. Thus, it is required to predict full-field information using deep learning [45].

2.3. Application of DL in building performance prediction

Related studies can be divided into two categories: 1) prediction of high-resolution wind fields by low-resolution wind fields using different methods which have been well validated for high Reynolds number two-dimensional flow prediction with high accuracy [46]. This type of research originates from data assimilation in meteorology, where low-resolution observations are fused with high-resolution forecast data. The high-resolution forecast data provide the background distribution, and the low-resolution observations are used for bias correction. For example, Kim et al. (2021) used deep learning to interpolate discrete wind environment measurement points around buildings to accurately predict wind speed values in unmeasured areas and obtain complete wind field information Kim et al. (2021) [47]. 2) Using a deep generative model to generate the full information of the high-resolution environmental field from nothing. This kind of model is difficult to train, and the model accuracy is more sensitive to the training data. For example, Tanaka et al. trained the convolutional neural network (CNN) to predict the wind distribution around a building complex. Their training set was obtained by a self-developed urban generation tool combined with batch CFD simulations. [48]. Musil et al. (2019) proposed a generalized flexible approximation model based on residual convolutional neural networks (ResNet) for near-real-time prediction of 3D steady turbulence, which could provide immediate iterations at the early stages of building design Musil et al. (2019) [49].

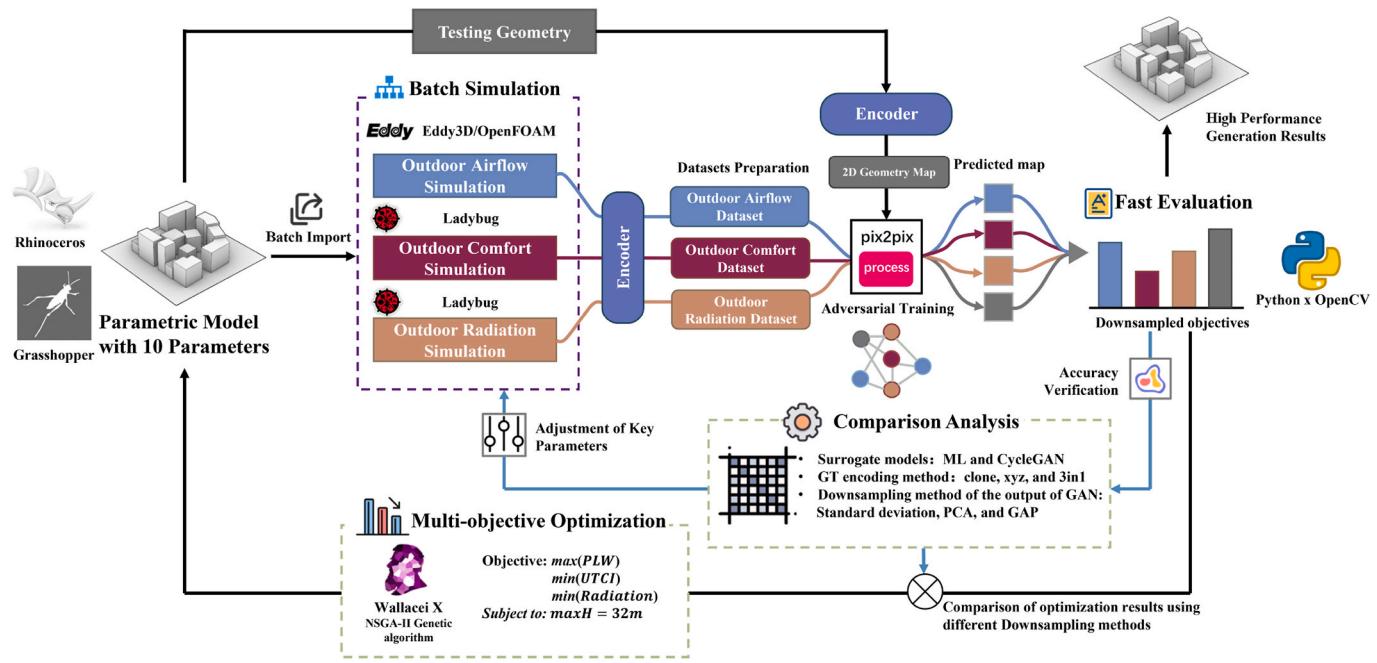


Fig. 1. Research workflow.

As one of the most popular DL method, generative adversarial networks (GANs) play a crucial role in computer vision. Some researchers have made significant progress with GANs to discover turbulent flows. GANs have been used in meteorology to generate high spatial and temporal resolution weather forecasts [50]. As a conditional GAN, pix2pix has gained much attention in building environments. It is based on an image-to-image translation that transforms an input image into an output pixel-by-pixel [51]. Conditional GAN allows the user to control the generated images, specifically allowing predictions for a particular condition (in this study, for a particular urban block). He et al. (2021) used pix2pix to predict the illuminance distribution in an indoor space He et al. (2021) [52]. Mokhtar et al. (2020) used pix2pix to learn the wind distribution around buildings, in which the training set also used the results of batch CFD simulations Mokhtar et al. (2020) [53]. Duering et al. (2020) combined pix2pix-based outdoor wind flow and a solar radiation prediction model with urban design and optimization systems to establish a workflow that reduces interoperability and computational cost barriers Duering et al. (2020) [54].

However, pix2pix is only an early tool for image-to-image translation. CycleGAN [55] and pix2pixHD [56] are improved versions of pix2pix, retaining conditional input. CycleGAN improves the data input format, and unlike pix2pix, cycleGAN can be trained using unpaired data. In addition, the special design of the network architecture in cycleGAN makes it possible to ensure that the original and generated images can be converted to each other without loss of information, enhancing the prediction's fidelity. The update of pix2pixHD for pix2pix makes it possible to input larger-size images (2048×1024 px), but the dimensionality of the data we obtain through the simulation engine is far from that large. If we employ image resizing function, the size mismatch needs to be filled by interpolation method, which could greatly reducing the reliability of the original data. In addition, a major obstacle lies in the hardware requirements. pix2pixHD requires a GPU with more than 11 GB of memories which hinders the generality and application of the method.

This study will use GANs to predict outdoor environments in urban blocks, which can capture global environmental details with high fidelity and support custom indicators selection to meet different site requirements. The GAN-based surrogate model combining with multi-objective optimization provides an efficient automated design

approach for the early design stages.

3. Methods

In this paper, a conditional GAN is applied to speed up the numerical simulation of the environmental performance of urban block, evaluate the pedestrian-level wind environment, outdoor thermal comfort, and cumulative solar radiation of the area of interest, and integrate them into a generative design workflow of multi-objective optimization based on a genetic algorithm. The workflow is shown in Fig. 1, in which the inner loop is the basis of fast performance evaluation, which core is the pix2pix adversarial training. The main process includes batch sampling of the parametric urban model, encoding, dataset preparation, adversarial training, prediction, downsampling and model adjustment. The outside loop realizes the environmental performance-driven generative design using pix2pix fast evaluation, and its core is multi-objective optimization based on a genetic algorithm.

(1) Environmental performance prediction based on pix2pix

A parametric urban block developed in the Rhino/Grasshopper platform that was used to create many geometries that are then imported into Ladybug tools (an open-source plug-in of Grasshopper for environmental performance simulation) for batch simulation. Through encoding the geometries and simulation results, training datasets for pix2pix were obtained. A generator was obtained through adversarial training, which allowed the input encoded geometry to obtain real-time environmental performance prediction results. By adjusting the model hyperparameters, the pix2pix generator was optimized.

In the encoder, we discussed three ways to encode the data to construct the input dimension adapted to pix2pix, named "clone", "xyz", and "3in1". In the model training section, we used cycleGAN to compare pix2pix models to evaluate the impact of model improvements on environmental performance prediction performance. We also compared the GANs with the classical surrogate model ANN to demonstrate the importance of predicting information on the spatial distribution of environmental indicators.

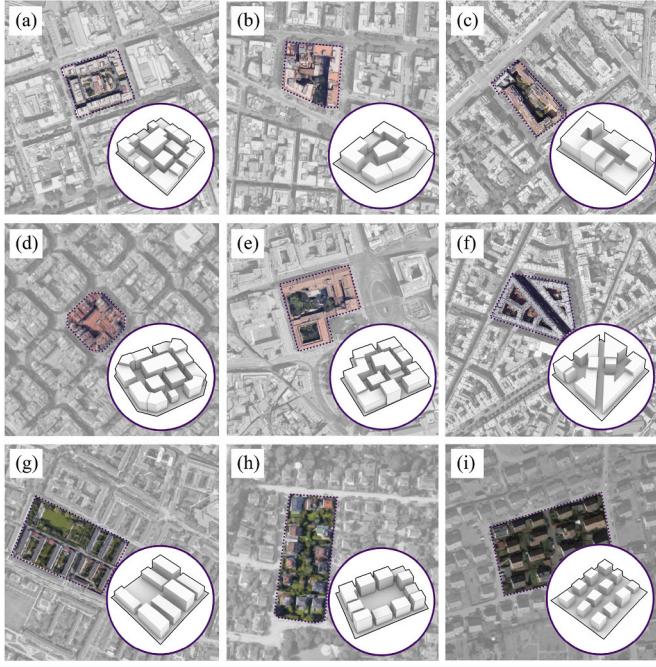


Fig. 2. Parametric urban block. The typical urban block pattern of (a) Rome: type 0; (b) Rome: type 1; (c) Paris: type 2; (d) Barcelona: type 3; (e) Rome: type 4; (f) Paris: type 5; (g) Munich: type 6; (h) Munich: type 7; (i) Heilbronn: type 8.

(2) Environmental performance-driven generation design based on a genetic algorithm

Wallacei X (a NSGA-II-based genetic algorithm tool in Grasshopper) was used for multi-objective optimization and controlling the parametric urban block. The optimization objectives include maximizing the

PLW velocity ratio, minimizing the UTCI, and minimizing the radiation. The fitness values of the genetic algorithm are predicted by pix2pix in real time, which ensures a rapid iteration of the optimization process. Finally, we obtained the Pareto solution set for the multi-objective optimization.

To verify the efficiency and effectiveness of the NSGA-II-based algorithm, we designed six optimization schemes. In which, we compared three downsampling approaches to transform the output of GANs into the objectives of the optimization algorithm, including global average pooling, principal component analysis, and standard deviation. Two multi-objective optimization algorithms are discussed, namely SPEA2 and hypE. In addition, we compared the results of GANs-based versus ANN-based optimization.

3.1. Parametric modelling of urban blocks

We have modelled 9 parametric prototypes block of based on general European urban morphology (Fig. 2). Our model can take as input any site boundary and generate buildings and public spaces. We have designed two basic generation logics (Fig. 3). The first one (type 0–5) is to prioritize the fitting of site boundary and generates an enclosed block morphology. Such urban blocks are common in Rome, Paris, and Barcelona. We set the urban block geometry back a certain distance from the site boundary and extrude the volume inside. The location and distance of the block openings and the height of each building in each generated geometry are randomized. In the second generative logic (type 6–8), we used Rhino/Grasshopper's plugin DecodingSpaces to divide the site and generate volumes in the subareas separately. Here, the buildings' length, width, and height are randomized. It allows us to obtain scattered strips or point blocks, a common style in Munich and Heilbronn. To increase the geometric diversity, the site boundary of each generated urban block can be randomly deformed and rotated. The control parameters for the block geometry generation are shown in Table 1.

We use six common urban morphological indicators to describe the

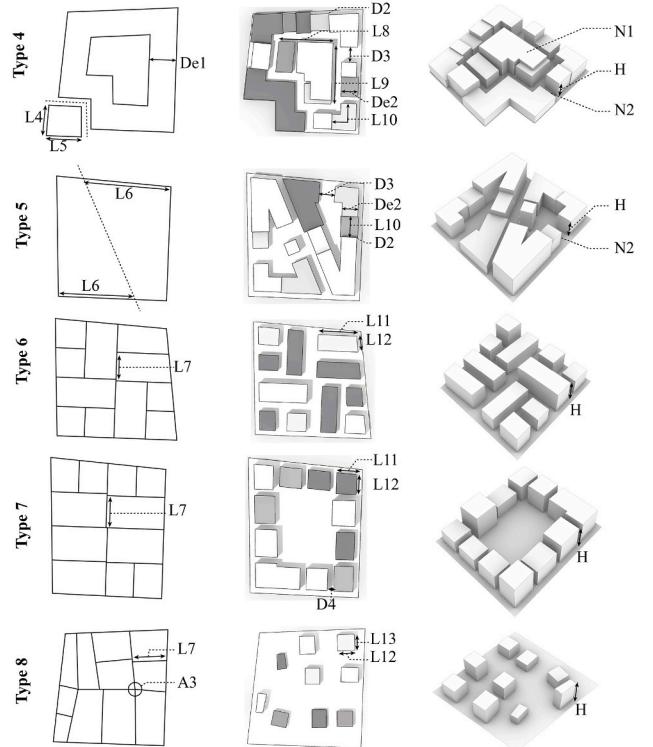
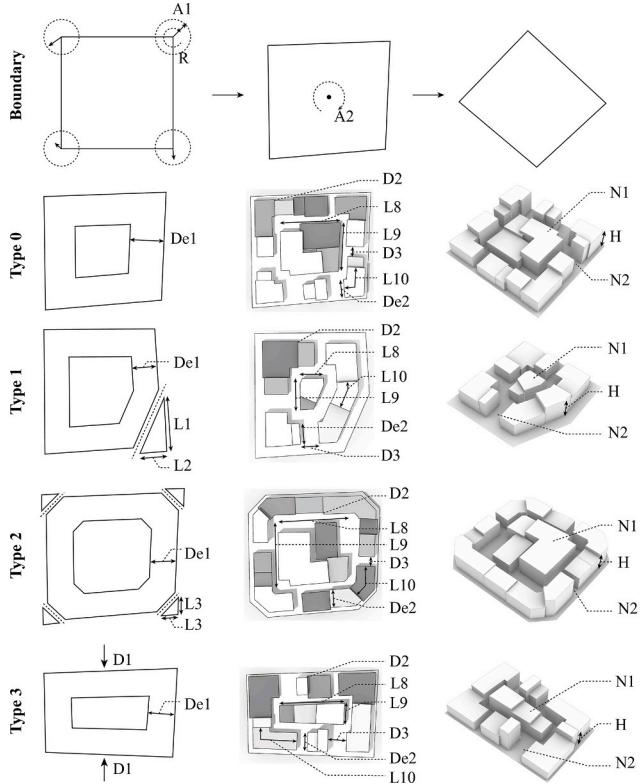


Fig. 3. Generation process of 9 types of the urban block.

Table 1
The control parameters of 9 types of the urban block.

Symbols	Range	Steps	Explanation	Types
Boundary parameters				
R	0.0-8.0	1.0m	Deformation radius of the corner point	Type 0-8
A1	0.0-360.0	0.1°	Deformation direction of the corner point	Type 0-8
A2	0.0-360.0	22.5°	Angle of the boundary rotation	Type 0-8
A3	0.0-25.0	0.1°	Angle of intersection point in site division	Type 8
De1	23.0-43.0	0.1m	Depth from boundary to interior façade	Type 0-4
D1	0-25.0	0.1m	Offset distance from the original boundary	Type 3
L1	20.0-60.0	0.1m	Length of the cut from the original boundary	Type 1
L2	20.0-60.0	0.1m	Length of the cut from the original boundary	Type 1
L3	12.0-30.0	0.1m	Length of the cut from the original boundary	Type 2
L4	20.0-30.0	0.1m	Length of the cut from the original boundary	Type 4
L5	20.0-30.0	0.1m	Length of the cut from the original boundary	Type 4
L6	60.0-120.0	0.1m	Length of subarea in site division	Type 5
L7	15.0-30.0	0.1m	Length of subarea in site division	Type 6
Planar parameters				
D2	0.0-4.0	0.1m	Dislocation distance of building façade	Type 0-5
D3	5.0-10.0	0.1m	Distance between buildings	Type 0-5
D4	3.0-6.0	0.1m	Distance between buildings	Type 7
De2	23.0-43.0	0.1m	Depth of exterior buildings	Type 0-5
L8	0-60.0	0.1m	Length for interior buildings	Type 0-4
L9	0-60.0	0.1m	Length for interior buildings	Type 0-4
L10	8.0-50.0	0.1m	Axis length of exterior buildings	Type 0-5
L11	20.0-60.0	0.1m	Length of building in subareas	Type 6-7
L12	16.0-22.0	0.1m	Length of building in subareas	Type 6-8
L13	16.0-22.0	0.1m	Length of building in subareas	Type 8
Volume parameters				
N1	0-5	1	Number of interior buildings	Type 0-5
N2	0-5	1	Number of openings in the block	Type 0-5
H	12.0-28.0	1.0m	Height of buildings	Type 0-8

Table 2
Urban morphological indicators.

Urban morphological indicators	symbol	Unit	Equation	Ref
Building coverage ratio	λ_P	%	$\lambda_P = (\sum_{i=1}^n A_{Pi})/A_T$	[9]
Floor area ratio	FAR		$FAR = GFA/A_T$	[57]
Average building height	H_{ave}	m	$H_{ave} = \frac{1}{n} \sum_{i=1}^n H_i$	[9]
The standard deviation of building height	H_{std}	m	$H_{std} = \sqrt{\frac{1}{n} \sum_{i=1}^n (H_i - H_{ave})^2}$	[9]
Building Shape Factor	BSF	m^{-2}	$BSF = S/V$	[58]
Frontal area ratio	λ_F		H/D	[9]

generated urban block, which is the building coverage ratio (λ_P), floor area ratio (FAR), average building height (H_{ave}), the standard deviation of building height (H_{std}), building shape factor (BSF), and frontal area ratio (λ_F). The equations are shown in Table 2.

3.2. Environmental performance simulation

3.2.1. Numerical simulation of the pedestrian-level wind (PLW)

The Eddy3D is an interactive interface based on the OpenFOAM CFD code, which is used to call OpenFOAM instructions visually and run the CFD model. OpenFOAM plays a vital role in practice due to its advantages, including open-source and customizable features, and many studies have tested its accuracy [20].

The CFD simulation parameter settings include the configuration of the computational domain, boundary conditions, and grid size. This paper refers to the existing guidelines and best practices for urban flow simulations [62,63]. The Eddy3D was used to call OpenFOAM to create a virtual wind tunnel. The distance between the wind tunnel's inlet, side, and top surface was set to 5H, and the outflow direction was set to 15H, where H = 32 m, the maximum building height. The inlet boundary condition was the velocity inlet boundary. The wind speed profiles were simulated using logarithmic law, where the reference height was set to z = 10 m. The reference wind speed is 5.25 m/s. This value is taken from

the wind speed corresponding to the prevailing wind direction in the city of Rome. The wind direction was westerly for all simulations because fixing the incoming flow direction helps reduce the input information to the GAN. During the data preparation, the site boundary was rotated randomly, thus preparing the dataset for each angle of inflow. Considering the high-density urban environment, the ground roughness length z_0 was set to 0.45. The atmospheric boundary layer thickness Z_g was set to 550 m. The top and side boundaries were set to symmetry boundary conditions so that the velocity component of parallel flow in the average direction of the boundary was zero. All surfaces were defined as no-slip walls, and the outlet boundary was defined as the pressure outlet boundary with a static pressure of 0.

We generated a hexahedral unstructured grid with an expansion ratio of 1.1 between two continuous grids. The minimum grid thickness around the building was set to 0.1 m. A monitoring surface was set at 1.5 m from the ground. The monitoring surface of 200 m * 200 m was divided into 65536 uniformly distributed monitoring points. The simulations were based on the 3D steady Reynolds Averaged Navier–Stokes (RANS) equations and the standard k-e turbulence model was used. The stop condition for solution convergence was a residual less than 10^{-4} .

In order to evaluate the sensitivity of CFD parameter settings, experimental data is required for comparison. AIJ provides wind tunnel test data on its website [59]. These wind tunnel tests were carried out to a high standard. The results of these experiments contribute to CFD validation. This study used AIJ wind tunnel test data to validate the simulations. Besides, we compared three different grid sizes' effects on computational accuracy and efficiency. The experimental validation and grid sensitivity analysis are presented in Appendix 1.

3.2.2. Simulation of the annual cumulative solar radiation (Radiation)

Studying the solar radiation that urban blocks receive is essential because it affects building heat gain, energy consumption and the heat island effect. Assessing solar radiation in urban blocks also provides the basis for building-integrated photovoltaics. The morphology of an urban block greatly influences the solar radiation received. This study used Ladybug to create a sky model and simulate the annual cumulative solar radiation (Radiation). Ladybug's solar radiation simulator uses

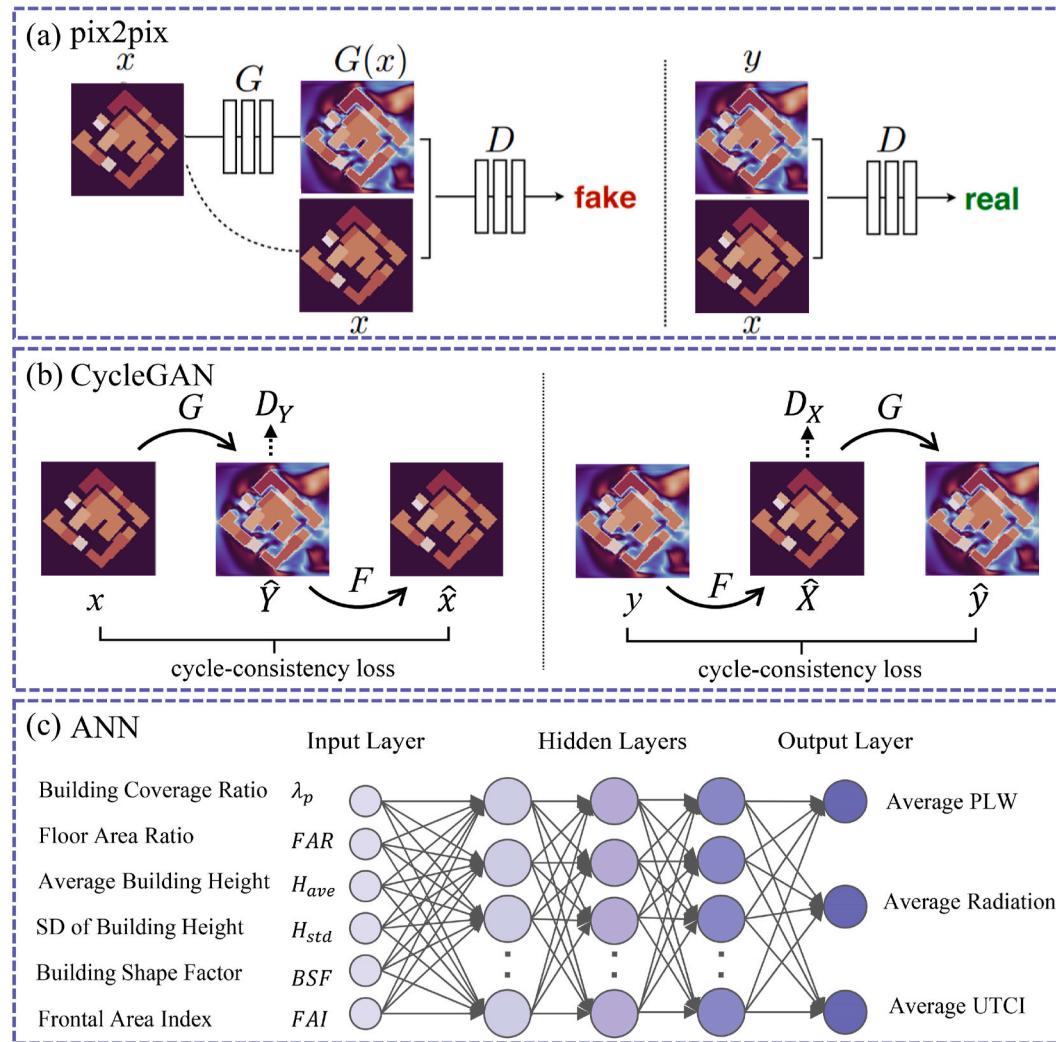


Fig. 4. The proposed three surrogate models (a) pix2pix [51], (b) CycleGAN [55], (c) ANN.

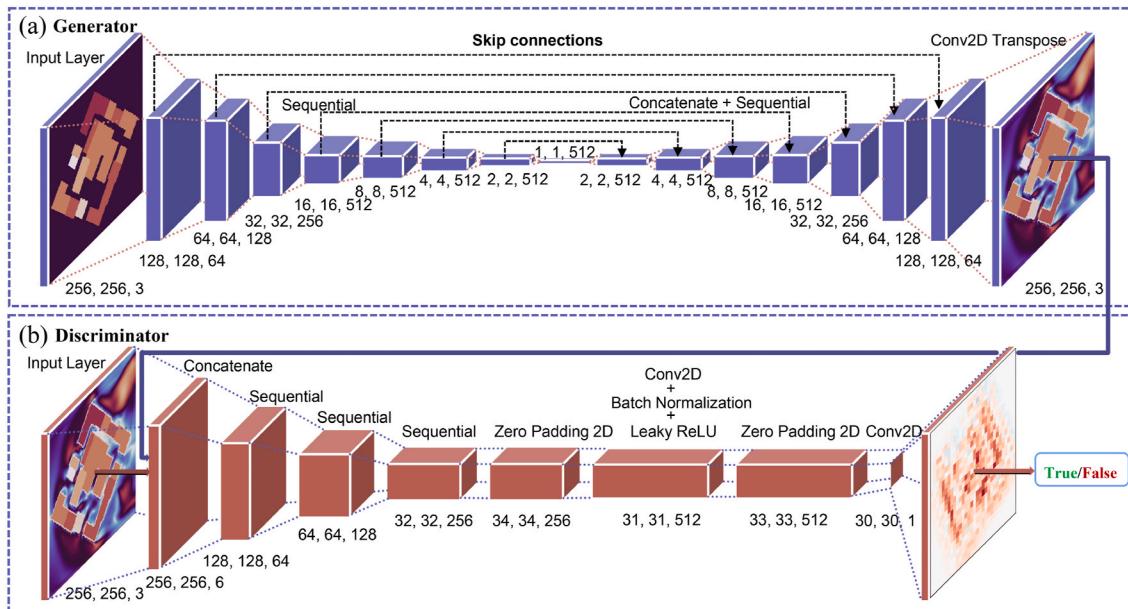


Fig. 5. Pix2pix model architecture (a) Generator, (b) Discriminator.

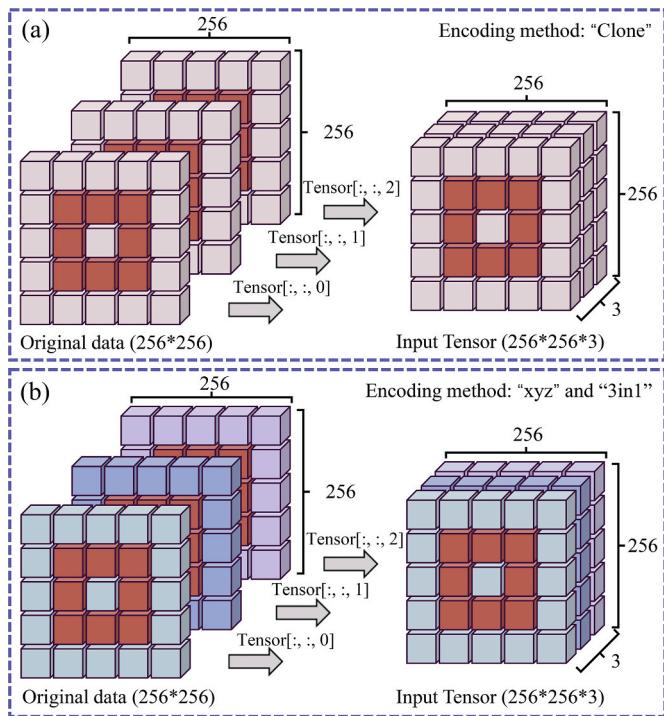


Fig. 6. The proposed encoding method (a) “clone” method, (b) “xyz” and “3in1” method.

Radiance software as the simulation engine, which has been subjected to long-term validation in practice. Geographical information of the simulated sites, and direct and diffused solar radiation are needed as inputs. The meteorological data used the epw file of the city of Rome. The calculation area was set to be the same as for the wind environment simulation. The calculation area was divided into a uniform grid of 256*256 for monitoring the simulation results.

3.2.3. Simulation of the universal thermal climate index (UTCI)

As a newly developed human thermal comfort index, the universal thermal climate index (UTCI) summarizes the interaction of environmental temperature, wind speed, humidity, long wave and short wave radiant heat flux. This assessment is based on the human body’s

physiological response, simulated using a thermophysiological model [60]. Krzysztof Blazejczyk summarizes the evaluation results of previous thermal comfort indexes and compares them with the UTCI. He believes that the UTCI can better describe the temporal changes in thermal conditions than other indexes and express the subtle differences in the intensity of meteorological stimuli [61]. The calculation formula of the UTCI can be expressed as follows:

$$\text{UTCI} = T_{amb} + f(T_{amb}, T_{MRT}, U_{wind}, p_{vapour}) \quad (1)$$

where T_{amb} is the ambient temperature, T_{MRT} is the mean radiant temperature, U_{wind} is the wind velocity, and p_{vapour} is the vapour pressure. The UTCI can reach a state of “no thermal stress” between 9 and 26 °C [51].

In this study, Ladybug was used to calculate the summer UTCI of the parametric urban block. The meteorological data for the simulations

Table 3
Optimization schemes.

	Surrogate model	Downsampling method	Optimization algorithm
Scheme 1	pix2pix	GAP	NSGA-II
Scheme 2	pix2pix	PCA	NSGA-II
Scheme 3	pix2pix	STD	NSGA-II
Scheme 4	pix2pix	GAP	SPEA2
Scheme 5	pix2pix	GAP	hypE
Scheme 6	ANN	–	NSGA-II

Table 4
Multi-objective genetic algorithm settings.

Population	
Generation Size	30
Generation Count	20
Algorithm parameters	
Crossover Probability	0.9
Mutation Probability	1/0.9
Crossover Distribution Index	20
Mutation Distribution Index	20
Random Seed	1
Simulation parameters	
Number of Genes	207
Number of Values	77385
Number of Fitness Objectives	3

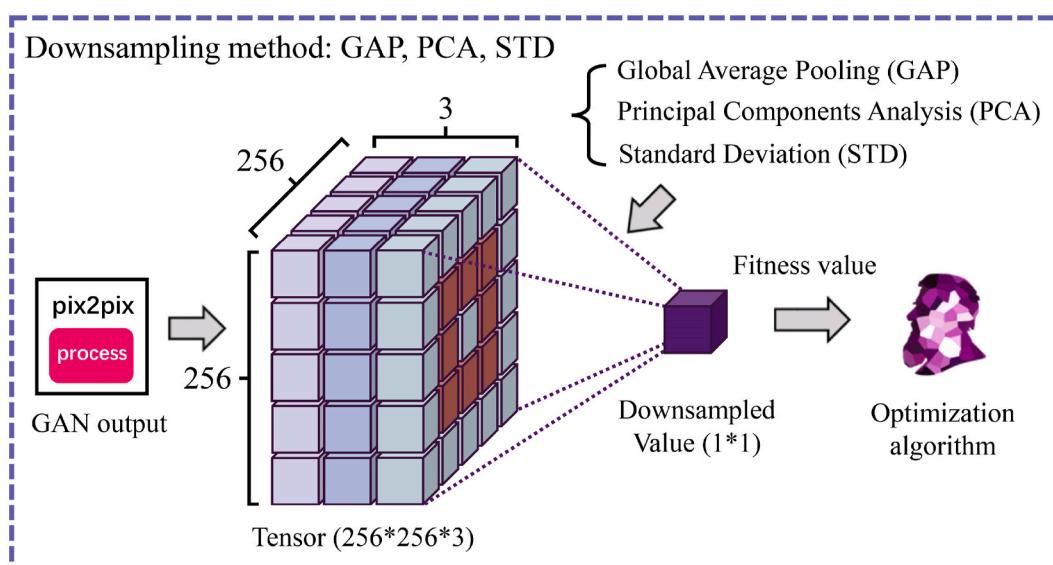


Fig. 7. The downampling methods.

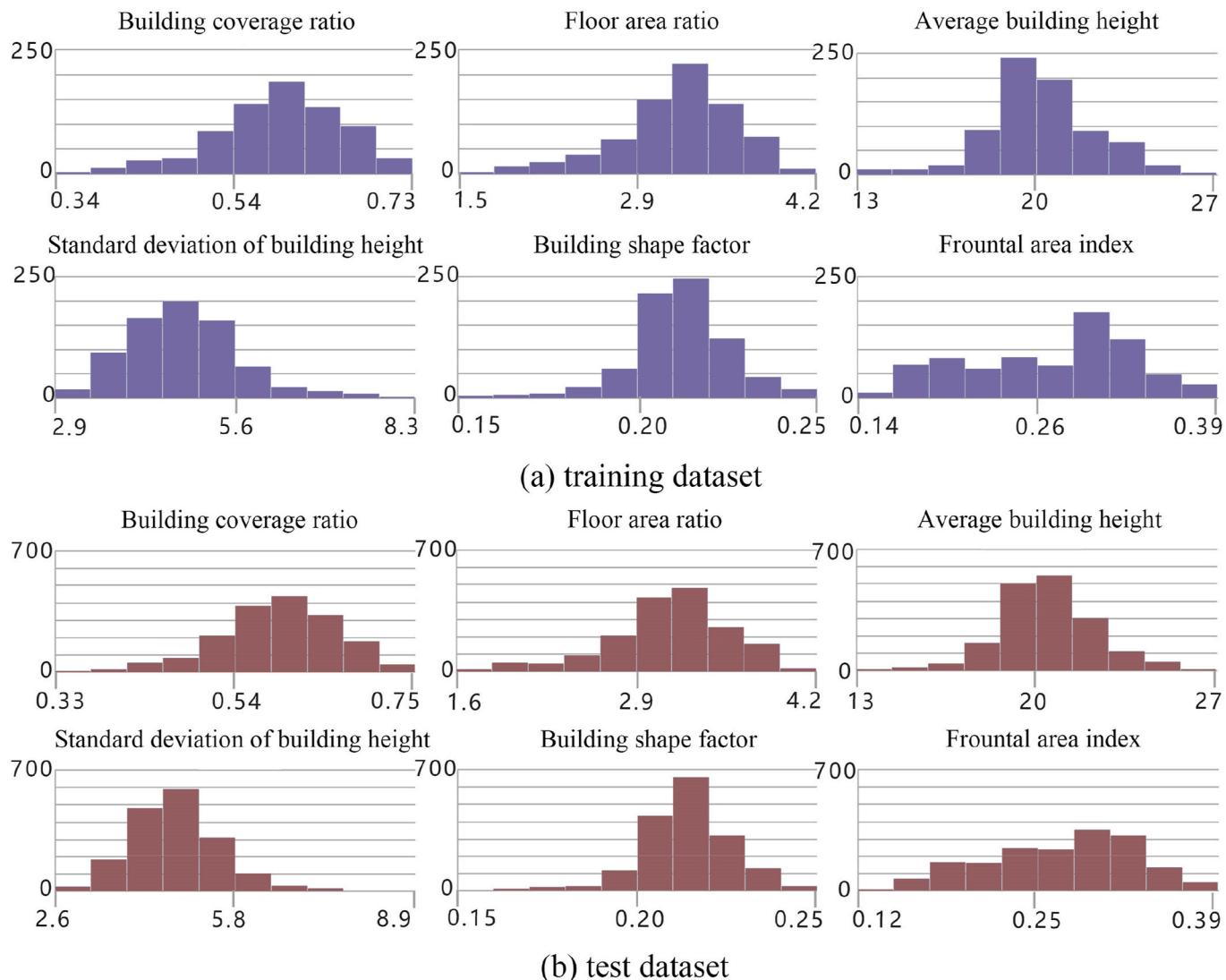


Fig. 8. Data distribution of the training and test dataset.

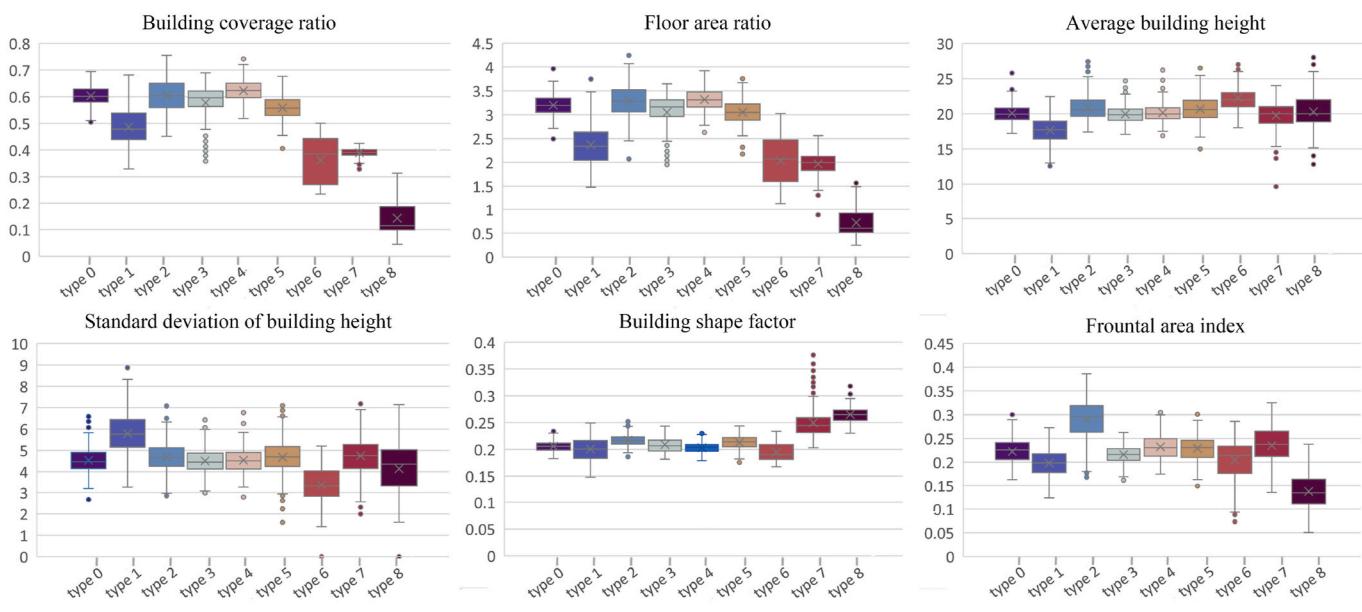


Fig. 9. The morphology difference of the nine types of urban block.

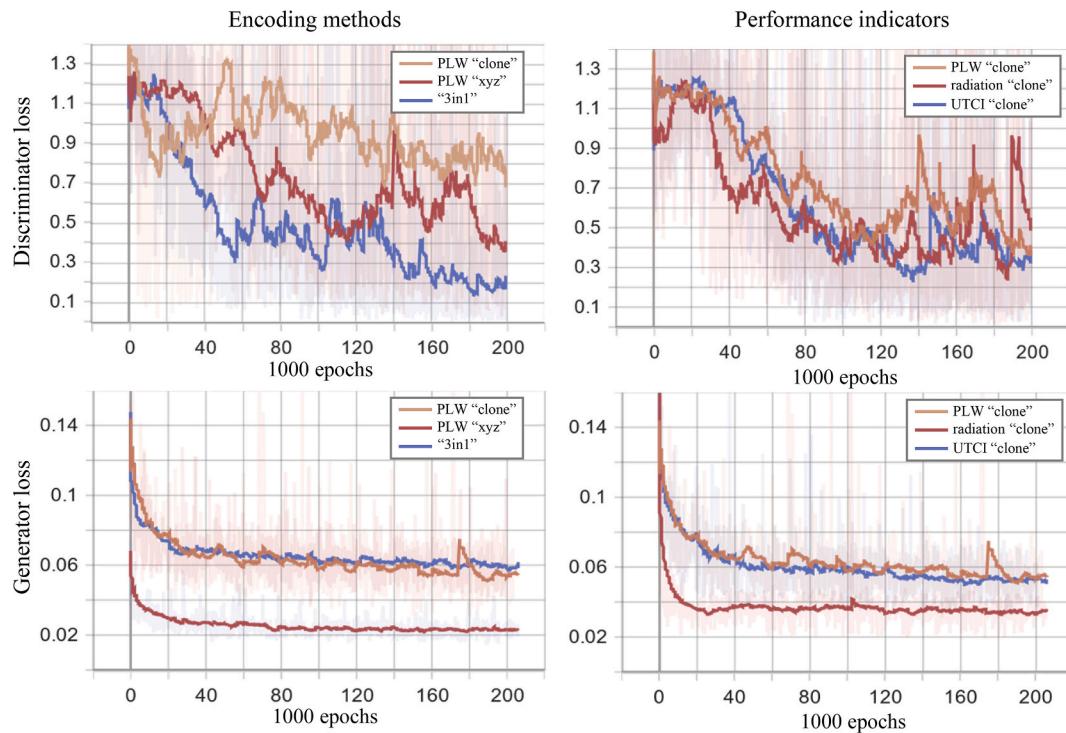


Fig. 10. Training loss curves.

were performed using the epw file for the city of Rome, and the date chosen for the simulations was July 21, the hottest day of the cumulative year in the city of Rome. The reason for using a typical day for the simulations is that the Mean Radiant Temperature (MRT), a key term in calculating UTCI, requires too much computational memory to support a year-round range of calculations. The average air temperature of the typical day, the calculated MRT distribution and the results of the wind simulation were used as inputs into the UTCI calculator. The monitoring points were set at 1.5 m from the ground in the same array of points as the wind and radiation simulations.

3.3. Training the generative adversarial network

As one of the most promising directions of artificial intelligence, GANs attracted significant attention as soon as they were developed. The traditional GAN aims at randomly generating images, which involves an unsupervised architecture. It consists of two groups of independent networks, which are adversarial targets. In the training process, discriminator (D) will receive the actual and false data generated by generator (G). Its task is to determine whether an image belongs to the true or false data. In order to train the generator, the model must adjust the parameters on both sides simultaneously. If D is correct, G parameters need to be adjusted to make the false data more real; if D is wrong, D parameters need to be adjusted to avoid similar judgement errors. Training continues until the two are in balance. In other words, they reach Nash equilibrium. Through the continuous training of D and G, generated images gradually become highly similar to natural images. To add more restrictions to the generation and enhance the application potential of GANs, conditional GANs appeared. Later, the pix2pix network was developed. A "real image and its condition" need to form a pair of training samples to build a training set. It can not only learn to generate images similar to natural images but can also learn the corresponding relationship between authentic images and their conditions. This method of image generation is called image-to-image translation.

In this paper, we compared pix2pix and a modified version of pix2pix, CycleGAN, with the classical supervised regression model for predicting PLW, Radiation, and UTCI. The models used in this paper are

shown in Fig. 4. Pix2pix feeds condition x, i.e., building information, into a generator G, which predicts the environmental data G(x), and a discriminator is used to determine whether G(x) matches the condition x or not. The output is true when the predicted environmental field matches the input building, and the spatial distribution of environmental indicators is close to the ground truth (Fig. 4(a)). CycleGAN uses the generator-discriminator architecture of pix2pix, and the update lies in setting an additional converter F. The predicted environmental field is generated after the building information x is input to the generator and then restored to x by the converter F. Training is completed means that the building information can be transformed with high accuracy with the environmental field data (Fig. 4(b)). ANN is one of the most commonly used surrogate models, and we build six morphological indicators for describing the block morphology and as the output of ANN, in which supervisory signal is obtained by averaging the environmental field.

The pix2pix generator and discriminator architecture are shown in Fig. 5. The generator of pix2pix is a modified U-Net which consists of an encoder (downsampler) and decoder (upsampler). Each block in the encoder is Convolution - Batch normalization - Leaky ReLU. Each block in the decoder is Transposed convolution - Batch normalization - Dropout (applied to the first 3 blocks) - ReLU. There are skip connections between the encoder and decoder (as in the U-Net). The discriminator in the pix2pix is a convolutional PatchGAN classifier—it tries to classify if each image patch is authentic or not. Each block in the discriminator is: Convolution - Batch normalization - Leaky ReLU. The output shape after the last layer is (batch_size, 30, 30, 1). The discriminator receives 2 inputs: the input image and the target image (the ground truth), which it should classify as true; the input image and the generated image (the generator's output), which it should classify as false.

We used normalized values (building height, wind speed, radiation intensity and UTCI) for encoding instead of creating a mapping of values to RGB. The purpose of normalization is, on the one hand, to facilitate feature extraction in the GAN, and, on the other hand, the normalized model can be applied to other regions by simply inverse normalizing the GAN output using local data. Building heights were normalized using



Fig. 11. Predicted results of three performance indicators for the “Clone” and “3in1” encoding method.

the set maximum height. The pedestrian level wind speed was normalized using the local prevailing unaffected wind speed of 5.25 m/s from the meteorological data source, and the solar radiation was normalized using the local unaffected maximum ground received solar radiation of 1267.054844 kWh. UTCI was normalized using the local design day unaffected 1.5 m height human UTCI of 30.802856° Celsius. The colours of the training data in the paper were visualized for the normalized values and would not affect the training performance.

The input of pix2pix is defined as a 256*256*3 third-order tensor, while our training data is a 256*256 two-dimensional matrix. In order to match the input data with the pix2pix architecture, we designed three encoding methods: “clone”, “xyz”, and “3in1” (Fig. 6). The “Clone” method means that the 256*256 training data are copied in thirds and superimposed in the depth direction of the tensor. This method applied to all data encoding, including the input geometry, PLW, radiation and UTCI. “Xyz” method was used for PLW. In the data preparation, we obtained the scalar wind field around the buildings at 1.5 m height and the vector wind field in xyz direction. The data format for each direction of component of wind velocity was a 256*256 two-dimensional array, and we superimposed all three in the depth direction. The “3in1” method discusses the superimposition of PLW, radiation and UTCI in the

depth direction into a 256*256*3 third-order tensor, which has the advantage of using a pix2pix model to train three targets simultaneously, reducing the training time and model size. It is worth mentioning that the “clone” and “xyz” methods both construct the tensor and then normalize it. The “3in1” method uses normalization before construction because of the inconsistent ranges of the three layers of data.

We used the TensorFlow deep learning framework to train the above networks. A generator with excellent performance can be developed by continuously adjusting the hyperparameters. Finally, the trained generator and data encoding method were deployed to the Rhino/Grasshopper platform via the GH_CPython plug-in.

3.4. Multioptimization based on a genetic algorithm

Using the Wallacei X plug-in, the surrogate model was integrated into a generative design workflow, and the rapid performance evaluation was combined with the visual multi-objective optimization process to evaluate the design strategy we could take. Wallacei X has a built-in integration of NSGA-II (Non-Dominated Sorting Genetic Algorithm II), a multi-objective optimization algorithm widely used in optimization

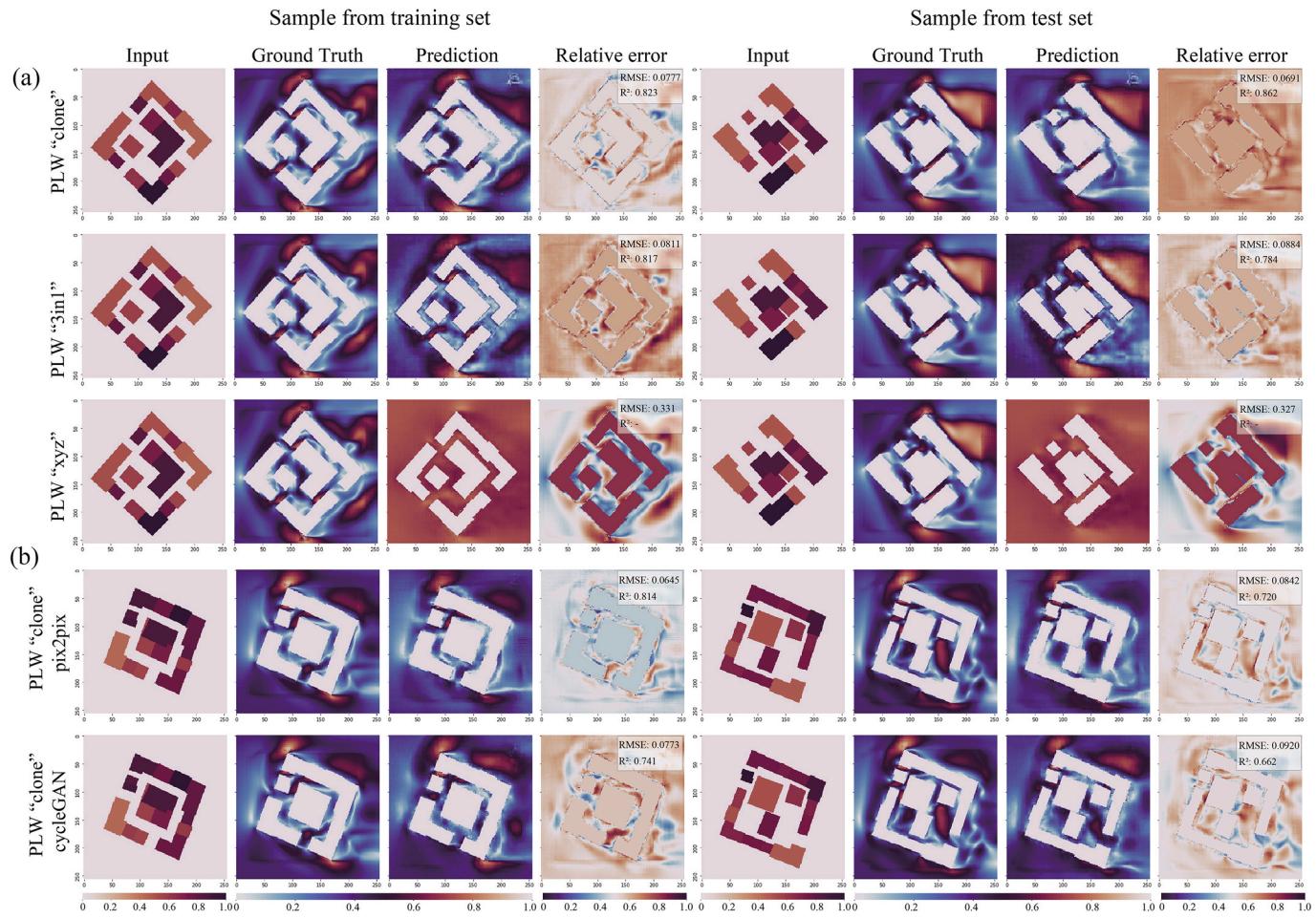


Fig. 12. Comparison of different methods for predicting PLW (a) Comparison of the three encoding methods (b) Comparison of the performance of pix2pix and CycleGAN.

Table 5
Performance of all training models.

Performance indicators	PLW				Radiation				UTCI			
	Training set		Test set		Training set		Test set		Training set		Test set	
Dataset	RMSE	R ²	RMSE	R ²	RMSE	R ²	RMSE	R ²	RMSE	R ²	RMSE	R ²
Evaluation indicators												
pix2pix "clone"	0.16	0.75	0.17	0.70	56.9	0.92	56.2	0.92	0.13	0.86	0.24	0.80
pix2pix "3in1"	0.19	0.73	0.19	0.68	66.2	0.96	61.4	0.93	0.15	0.77	0.22	0.71
pix2pix "xyz"	0.63	0.02	0.56	0.05	-	-	-	-	-	-	-	-
CycleGAN "clone"	0.20	0.60	0.21	0.58	151	0.85	140	0.88	0.28	0.61	0.39	0.50
CycleGAN "3in1"	0.23	0.64	0.16	0.40	103	0.87	101	0.84	0.36	0.53	0.36	0.54
ANN	0.14	0.78	0.15	0.75	30.1	0.96	37.3	0.93	0.14	0.78	0.17	0.72

cases. It is based on the following principle: First, N solutions are randomly selected to form an initial overall P in the search space. Its objective is to calculate the non-dominated level and the congestion distance. Using a fast non-domination sorting method, the solutions in P are classified into different non-domination levels, where solutions in the same level are non-dominated, and solutions in the higher level are dominated by at least one solution in the next lower level. Solutions with lower levels of non-domination and more considerable crowding distances are preferred in the selection process to maintain convergence and diversity.

Since the generator's output is a 256*256*3 3D tensor, there is challenging to process the output as a target value. Processing the output needs to consider maximizing the preservation of the spatial distribution

characteristics of the environmental indicators rather than simply taking the average value. We refer to the process of deep learning down-sampling and compare three downsampling methods to downsample 256*256*3 to a single value. Specifically, we used standard deviation (STD), principal component analysis (PCA) and global average pooling (GAP) methods for downsampling (Fig. 7). STD is a classical statistical method calculates standard deviation for all elements in a third-order tensor. The standard deviation reflects the homogeneity of the spatial distribution of PLW, radiation and UTCI. PCA is a popular method for dimensionality reduction and is commonly used for feature extraction. We used the PCA method to reduce the dimensionality of the output and extract the high-dimensional features of the distribution of environmental indicators. We hypothesize that using high-dimensional features

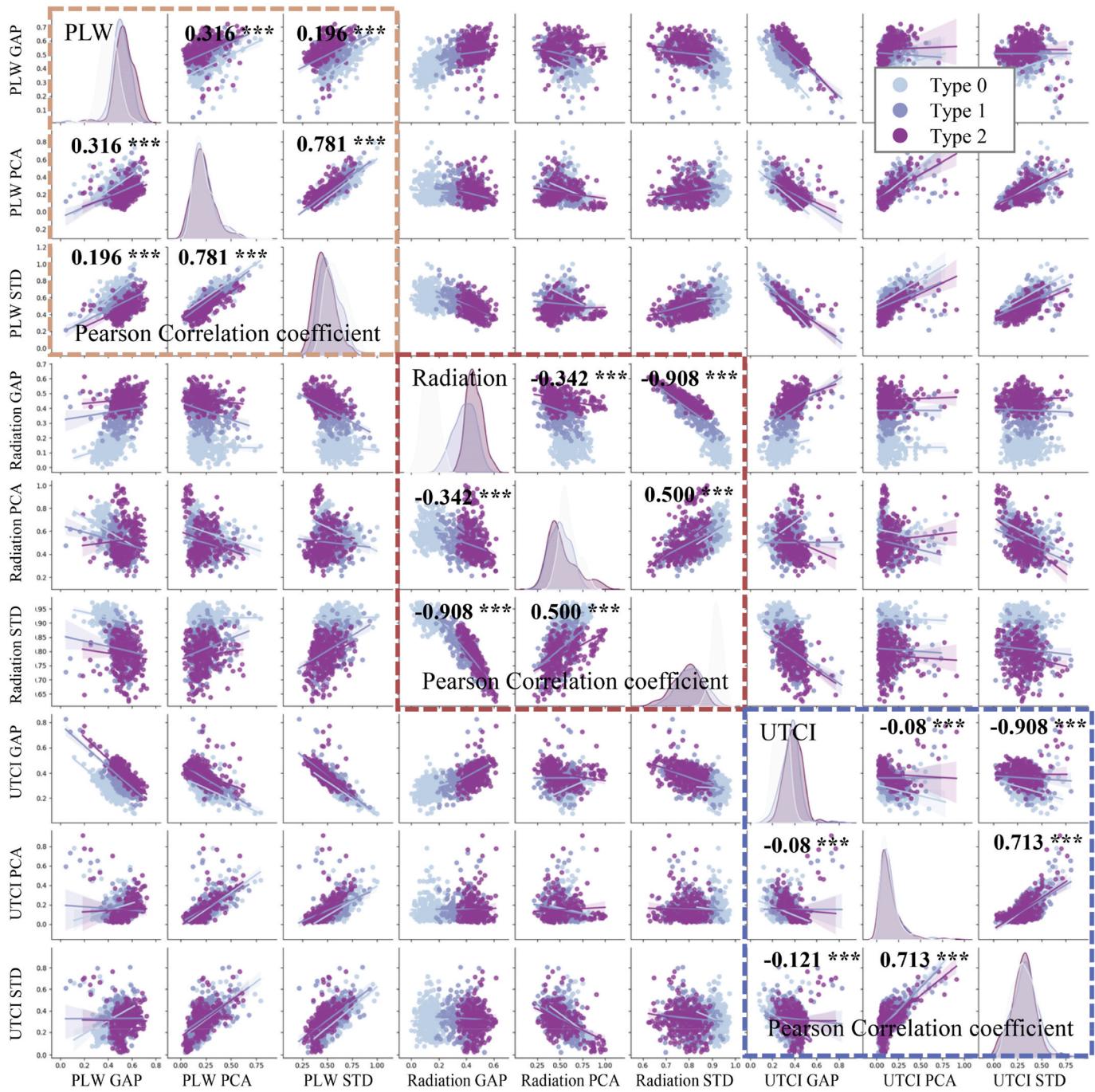


Fig. 13. Correlation analysis of downsampled objectives.

as targets may benefit the optimization process. The GAP method is used in the deep learning process to replace fully connected neural networks, and it is based on the principle of outputting a mean value for each feature map. Compared to fully connected networks, it does not require too many parameters and reduces the risk of overfitting in the dimensionality reduction process. GAP considers the spatial information in the tensor compared with the global mean. This study is the first to propose using downsampling to build a bridge between pix2pix output and genetic algorithms.

Different downsampling methods lead to different optimization directions. In the STD-based optimization case, we set minimizing three objectives to obtain the most homogeneous global environment. In the

GAP-based optimization, we maximized PLW, minimized Radiation and UTCI, increased urban block ventilation and reduced heat and discomfort. We compared the optimal set of solutions using the three downsampling methods. In particular, the target value of PCA output does not belong to the same interval as the performance indicators, and we do not yet know the relationship between the performance indicators after PCA and the original data. A statistical analysis of the three downsampling methods is needed to determine the optimization direction of PCA.

We designed six optimization schemes (Table 3), among which the NSGA-II algorithm based on the Wallace plug-in was compared using three downsampling methods. In addition, we compared two optimization algorithms SPEA2 and hypE of Octopus plug-in of Rhino/

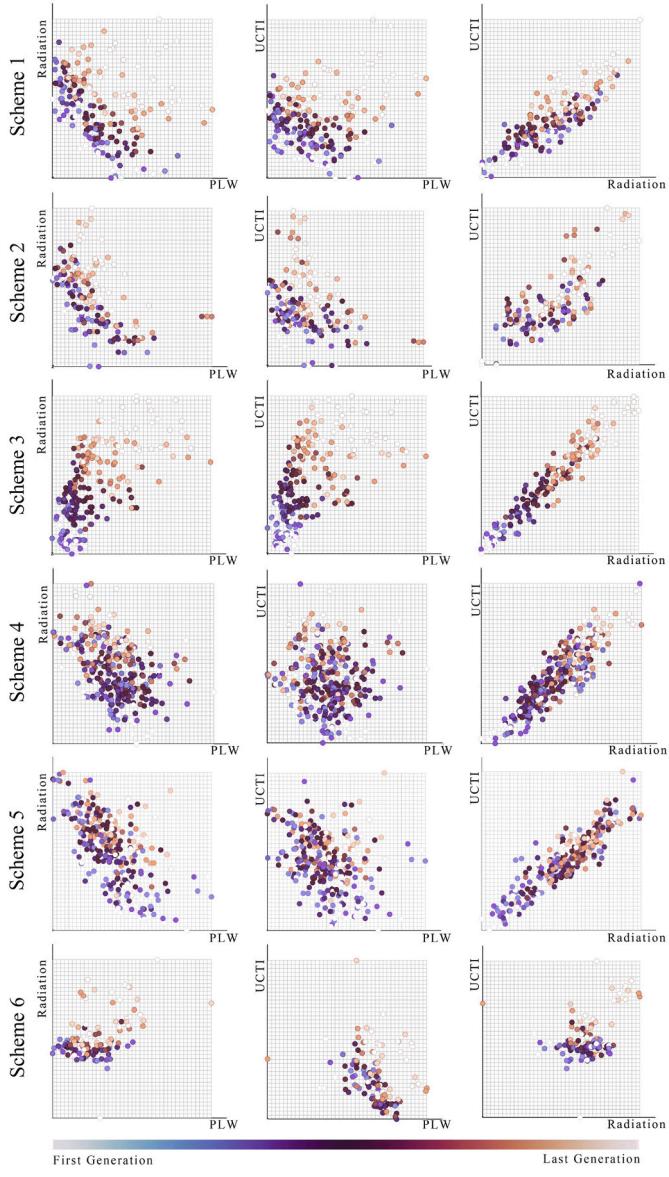


Fig. 14. Optimized solution space visualisation.

Grasshopper platform during the optimization process. To verify the superiority of surrogate models based on GANs, we compared the optimization of GAN with ANN algorithm as a surrogate model. The best model trained in section 3.3 was chosen for all schemes. The parameters of the optimizer for scheme 1 are shown in Table 4.

4. Results and discussion

4.1. Dataset description

We created 9 types of urban block and sampled 300 geometries from each type for simulation, and finally, excluding the anomalous results, we obtained 2665 samples. The samples were processed as paired data for training pix2pix and cycleGAN. All data was divided into a training set (1858) and a test set (807) by 70% and 30%.

The pix2pix preset data augmentation strategies include cropping, rotating and mirroring—due to the translation and rotation invariance of CNN in recognizing natural images. However, we consider the spatial distribution of environmental variables based on physical principles that

are only translation invariant and not rotation invariant. We changed the pix2pix image augmentation strategy by resampling the data to 286*286 and then cropping the resampled data four times. The final training samples were increased to five times the original training data, 9290. See Appendix 2 for more descriptions of data augmentation methods.

For ANN, we prepared tabular training data for the same samples. Parametric block morphology was used as features, including building density, floor area ratio, average building height, the standard deviation of building height, shape factor and frontal area index. According to relevant research practices, the predictor variables were set as the average of each of the three indicators.

Fig. 8 shows the distribution of the building morphology indicators for all samples in the training and test sets, and the results show that the distribution is uniform. It proves the reasonableness and randomness of the data set division. Fig. 9 shows the range of the morphology indicators of the nine types of urban block in the whole data set, and most indicators have significant differences between the types.

4.2. Pix2pix generation results and parametric sensitivity test

The loss curves of the optimal model training are shown in Fig. 10. GANs learn a loss that adapts to the data, while cGANs learn a structured loss that penalizes a possible structure that differs from the network output and the target image. The generator loss is a Mean Absolute Error (MAE) between the generated and target data. It allows the generated data to be similar in structure to the target data. The loss function of the discriminator requires the input of the real and generated data and shows the distance of the generated data from the real data.

Fig. 10 monitors the training process of the three encoding methods. All training was set to stop at 200,000 epochs. Overall, all generator and discriminator loss functions show a decreasing trend, and the “xyz” method has the lowest generator loss compared to the other two methods. However, considering that the discriminator loss introduces oscillations, this may lead to an increase in the instability of the “xyz” method. The generator losses of the “clone” and “3in1” methods are similar. However, the discriminator losses of the “clone” method are higher, which means that it may be more difficult to determine the authenticity of the generated images. We compared the learning process of the three performance indicators using the “clone” encoding method. The three performance indicators have similar discriminator loss, and the generator loss of radiation is significantly better than that of PLW and UTCI, implying a higher prediction accuracy of radiation.

Fig. 11 shows the prediction results of the samples in the training and test sets. The two encoding methods, “clone” and “3in1”, were compared. The relative error shows the difference between the prediction and the ground truth. All results are normalized, which makes it possible to apply the trained model to other climate regions by using different reference values for the inverse normalization. In the “clone” method, the pixel-by-pixel Root Mean Square Error (RMSE) of the predicted wind speed of the sample is 0.0556, and the coefficient of determination R^2 is 0.852, which shows that the predictions are structurally similar to the ground truth. Visualization result shows that the errors in the predictions mainly appear in the interior region and the wake flow. The prediction of radiation is relatively good, and the relative error shows less global error and uniformity in distribution. Regarding UTCI, although RMSE and R^2 show good results, the relative error shows the difference in local prediction. It may be because the calculation of UTCI involves PLW and MRT, and the model learning errors for PLW and Radiation may be superimposed in predicting UTCI. In the “3in1” method, the Radiation prediction has a clear advantage with a coefficient of 0.912 for the sample. Global predictions of PLW and UTCI also show an acceptable accuracy. The test set has a slightly lower sample overall than the training set, but the predicted data are still structurally similar to the ground truth.

Fig. 12 shows the different methods for predicting PLW. Fig. 12(a)

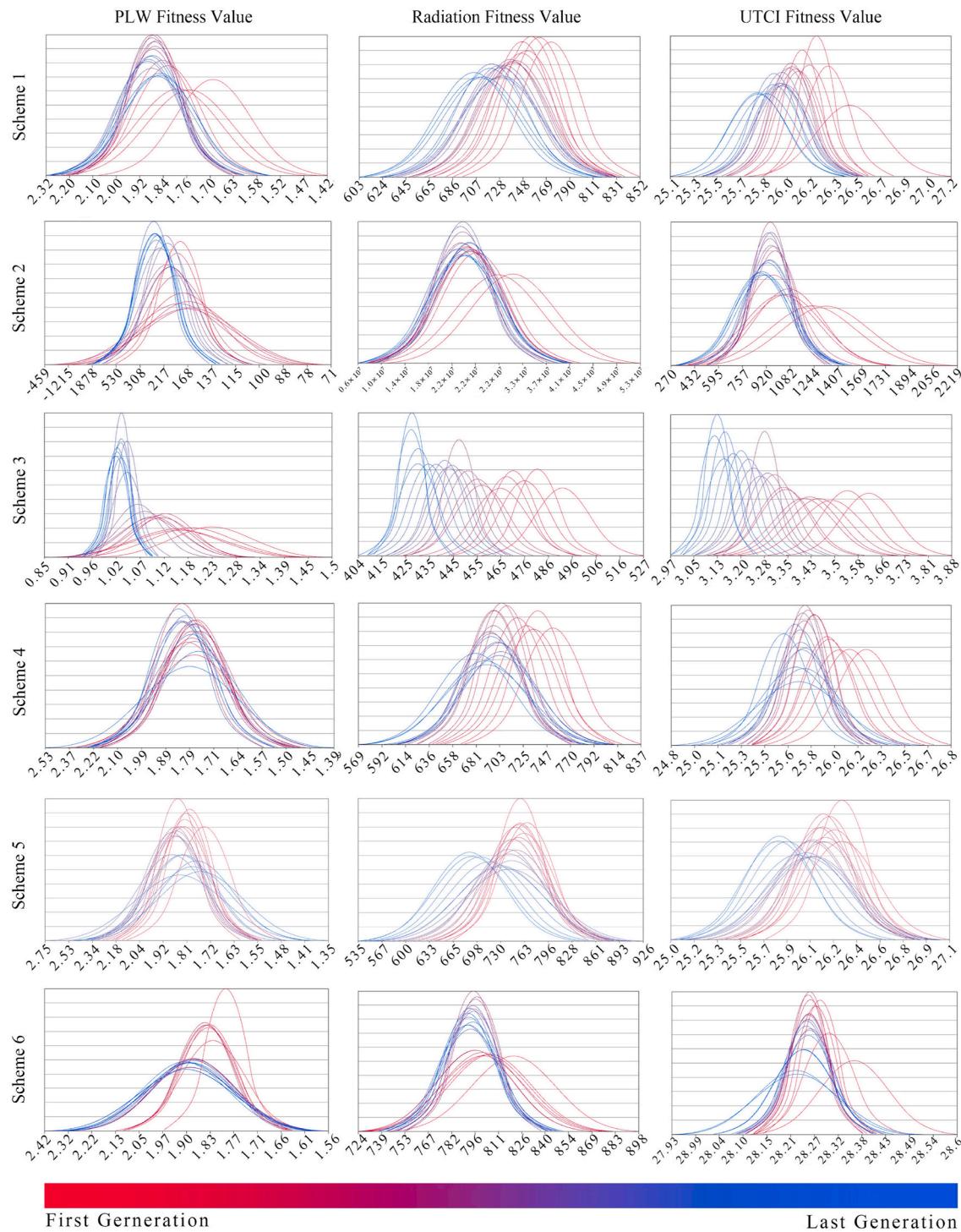


Fig. 15. The standard deviation of the objective during the optimization process.

compares the PLW prediction results of the training set and the test set using the “clone” method. The results show that the overall prediction accuracy is better using the “clone” and “3in1” methods. The encoding using the “xyz” method failed to learn the spatial distribution of PLWs in both the training and test sets. This result is contradicted by the best generator loss of “xyz” in Fig. 10. The possible reason is that the uniform encoding in “xyz” eliminates the spatial variability of the data. However, if the data are encoded separately as in the “clone” method, the “xyz” method will not be able to generalize across regions because the xyz-

directional sub-velocity does not have a useable reference wind speed in each region. Fig. 12(b) compares the prediction results of pix2pix and CycleGAN for PLW, which shows that the prediction performance of CycleGAN is not as good as that of pix2pix. Probably because although CycleGAN is an upgraded version of pix2pix, it seems more applicable to natural images. CycleGAN is designed with the strategy of transforming the input data into target data and back, and this transformation is more challenging in the face of predictions from physically controlled environmental fields. It also results in a training time of CycleGAN that is 2–3

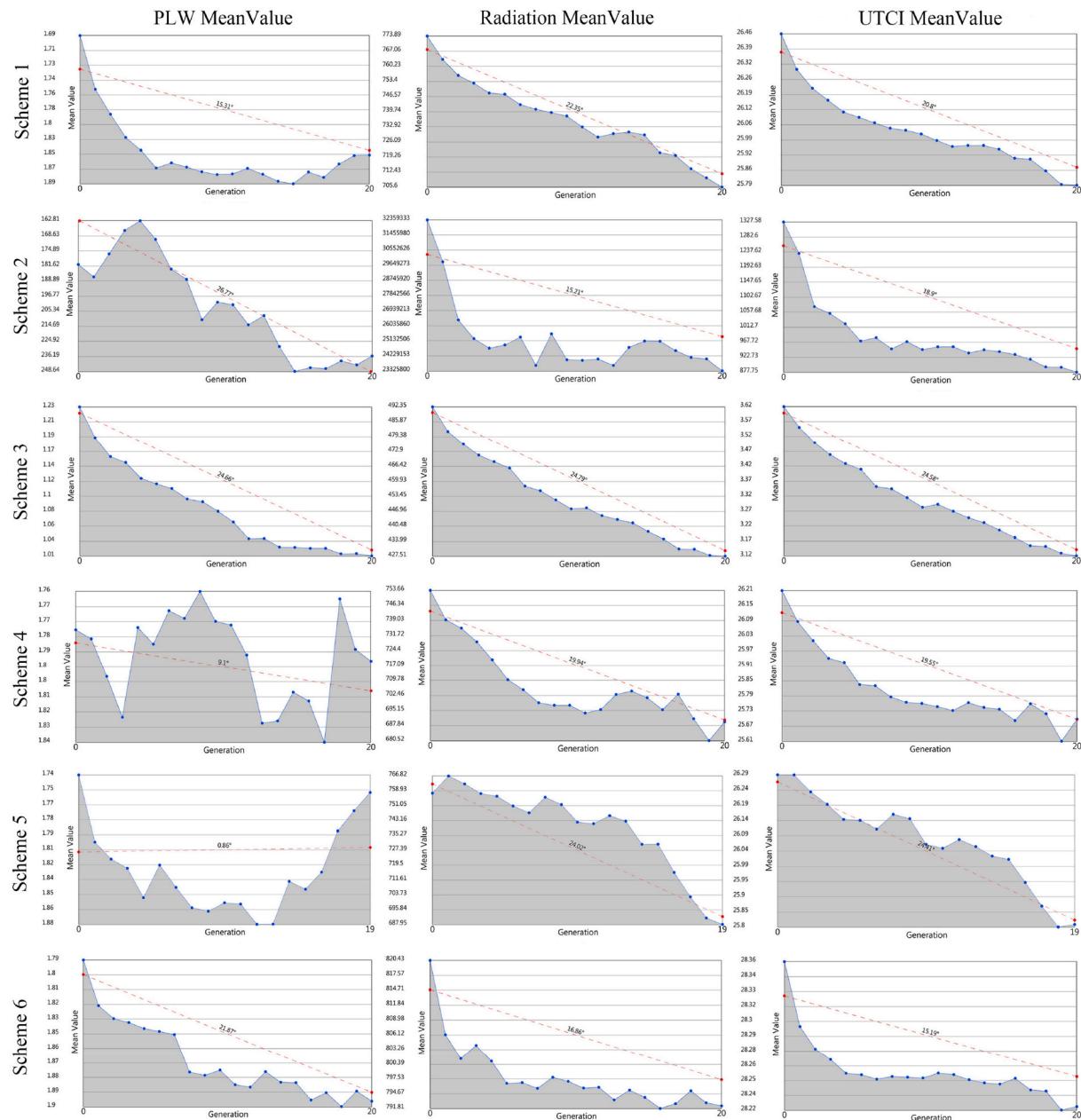


Fig. 16. The objective value during the optimization process.

times longer than pix2pix.

Table 5 shows the statistical values of all models in the training and test sets. The data in the table are the average of the pixel-by-pixel predicted differences for all samples in the dataset. The pix2pix model trained with the “clone” encoding method predicts PLW with an average R^2 of 0.75 in the training set and 0.70 in the test set and predicts radiation with an average R^2 of 0.92 in both the training and test sets. The same method predicts the UTCI with the average R^2 for the training set is 0.86 and the average R^2 for the test set is 0.8. The prediction performance using the “3in1” method is slightly lower than that of the “clone” method, which may be because compared to the “3in1” method, the three performance indicators of the “clone” method hardly affect each

other. In the “3in1” method, the three performances are considered a whole for convolution and feature extraction. However, the “3in1” method still has potential to be applied because its model size and training time are only 1/3 of the “clone” method. Unfortunately, the “xyz” method fails in learning the spacial distribution of PLW. The performance of CycleGAN is generally lower than pix2pix. One possibility is that setting the same training epoch as pix2pix limits the performance of CycleGAN. However, even with the same number of epochs, CycleGAN consumes more computational resources. The ANN results show higher accuracy in both PLW and Radiation predictions. However, ANN is only used to predict the mean value, and the prediction is much less difficult than predicting the spatial distribution of environmental

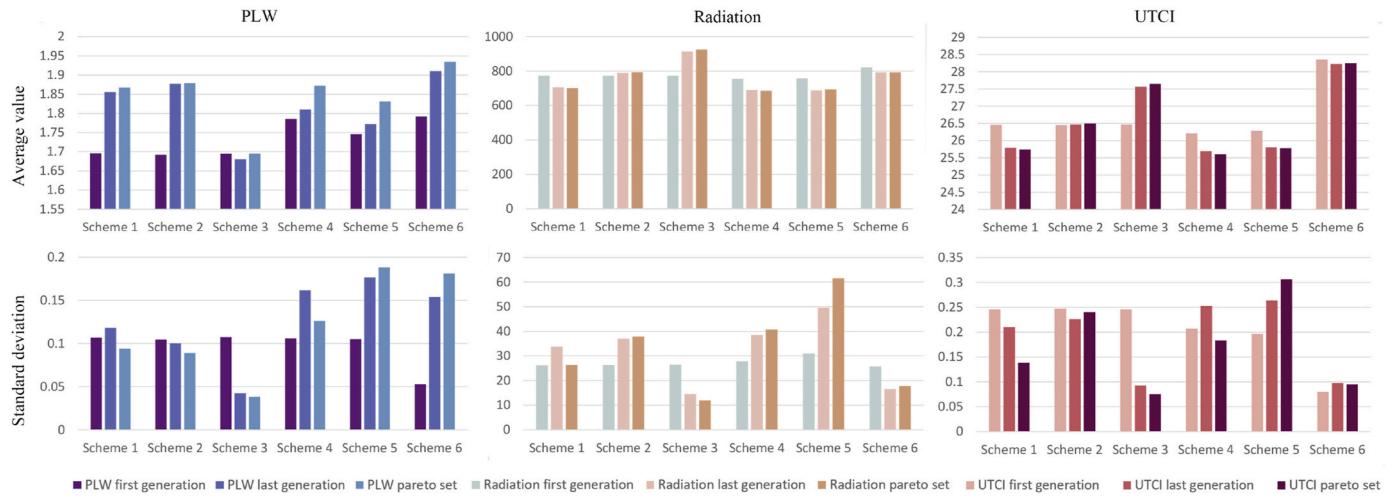


Fig. 17. Comparison of the optimization performance of different schemes.

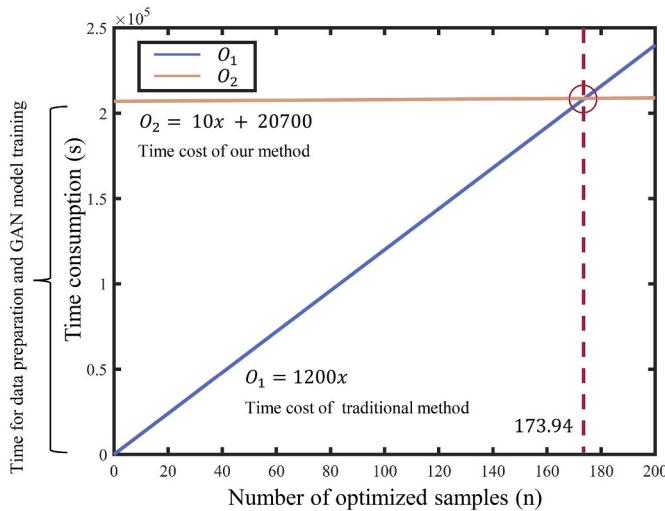


Fig. 18. Time consumption of the proposed method and traditional method.

indicators. In the next subsection, we will compare the results of pix2pix using the “clone” method with those of ANN as a surrogate model optimization.

4.3. Multi-objective optimization solution space of the genetic algorithm

We used three downsampling methods, GAP, PCA and STD, for the global environmental simulation results of PLW, radiation and UTCI in the whole dataset. The purpose is to determine the optimization direction of different downsampling methods. The calculated results are visualized by type. The example of type 0, 1 and 2 is shown in Fig. 13. The overall goal of the optimization case is to maximize PLW and minimize Radiation and UTCI, and we want to minimize the standard deviation of the data on this basis to reduce the differences in spatial distribution. For the three performance indicators, if PCA is positively correlated with the standard deviation, then the optimization direction of PCA is minimization, and if PCA is negatively correlated with the standard deviation, then the optimization direction is maximization. The three types of urban blocks show that this correlation is generally consistent across morphologies.

Fig. 13 shows the optimization direction for each scheme, and Fig. 14 shows the optimization results. Each point in the figure represents a

sample of the optimization process. The samples from the first to the last generation of schemes 1–5 are significantly different, and the samples of ANN-based scheme 6 are less distant in space for each generation. It may mean that the optimization performance of scheme 6 is insufficient.

Fig. 15 show the standard deviation of the objective during the optimization process. Each line represents a generation. The redder the colour, the closer to the initial generation, and the bluer the colour, the closer to the final generation. The highest point value of the curve's horizontal coordinate represents the average target values for this generation of samples. In schemes 1–3, the standard deviation of the samples in each generation gradually decreases as the last generation is approached, which means that the samples are more homogeneous among themselves. At the same time, there is a significant difference between the target values of the initial generation and the last generations' target values. Scheme 3 has the best optimization performance and proves the application value of the STD downsampling method. The standard deviation between samples in each generation increased as the target value was optimized in the optimization process of scheme 6. This suggests that it may not be appropriate to use ANN as a surrogate model for global optimization.

In Fig. 16, the trendline further shows the convergence of the genetic algorithm. Similar to the conclusion of Fig. 15, the optimization process of scheme 3 is the smoothest. Schemes 1, 2, 4, and 5 show different fluctuations. Among them, schemes 4 and 5 show the most drastic fluctuations, suggesting that the SPEA2 and hypE optimization algorithms do not apply to this study's optimization task. The fluctuations in the decreasing target values are mainly found in the optimization of PLW. The possible reason is that PLW is most sensitive to different urban block morphologies. In addition, the accuracy of the prediction model may also be an influencing factor.

We calculated the average and standard deviation of each sample of the initial and final generations and the Pareto solution for each optimization scheme, displayed in Fig. 17. The results show that most of the schemes are valid. For the three environmental indicators, scheme 1 increases the average value while reducing the standard deviation. It demonstrates the superiority of the NSGA-II combined with the GAP downsampling method. Scheme 2 performs better during the optimization of PLW but fails in Radiation and UTCI. The average and standard deviation of Radiation and UTCI were increased in the final generation. Scheme 3 optimises the standard deviation of all environmental indicators but does not successfully optimise the three target values. Schemes 4 and 5 achieve the optimization of the target values, but the optimization of the standard deviation is ignored. Scheme 6 achieves an obvious optimization of the average value of PLW from the first to the last generation, but at the cost of a dramatic increase in the standard

deviation. Meanwhile, Scheme 6 does not work well in optimizing Radiation and UTCI. The comparison between scheme 1 and scheme 6 demonstrates the necessity of using GAN as a surrogate model.

4.4. Time consumption of our method and traditional method

The devices used in this study include a local computer (Intel i7-12700 RTX3070Ti) with a Microsoft Azure virtual machine (D64ds_v4). With the block scale and simulation parameter settings in this study, the computation time for a single PLW simulation is 7–11 min, a single UTCI simulation takes 5–8 min, and a single radiation simulation is about 3–5 min. The complete simulation of the three indicators takes about 20 min. We performed simulations for 2700 samples, which took about 54000 min, or 37.5 h. A single training of the pix2pix model for 200,000 generations took about 4 h, and the time for calculating hyperparameter tuning and data processing cost a total of 20 h. The total time spent on data preparation and training is 57.5 h.

Once the training was completed, the training data was deployed to the Rhino/GH platform, and a single prediction of three indicators took 5–10 s, which means a 120–140 times speedup of the environment simulation. In this study, the genetic algorithm was set to optimise 20 generations, with 30 samples per generation, for a total of 600 evaluations, and the optimization process took about 3000s–6000s, or 0.8h–1.7h when using the GAN-based surrogate model. While 200 h were required to optimise the same number of samples using the numerical simulation. The total time consumed far exceeds the total time for training and optimizing the GAN-based surrogate model.

A simple linear model can be constructed to calculate the time balance between the two methods (Fig. 18). For the simulation-based optimization process, $O_1 = 1200x$, and for the GAN-based optimization process, $O_2 = 10x + 207000$, where x is the number of optimized individuals, and O is the total time consumption of the optimization process in s. The parameters are derived from the records of this study. When the number of optimized samples reaches 173.9, the time consumption of both solutions is equal. Until then, simulation-based optimization has a time advantage, and when the number of optimized samples exceeds 174, the GAN-based surrogate model has a time advantage. The real situation may be more complicated, including that the simulation-based optimization process may face the difficulty in memory release and the resulting gradual slowdown of the performance simulation. In addition, the trained model performs generalisation and can be used for urban blocks with similar climatic conditions without additional data preparation and training.

5. Conclusion and future works

This paper presents an environmental performance-driven generative design framework using a generative adversarial network to replace environmental performance simulation processes, including the pedestrian-level wind, thermal comfort, and accumulated solar radiation. The proposed method avoids long-term meshing and simulation, and enables rapid prediction and evaluation of the environmental performance of urban blocks. At the same time, combined with a genetic algorithm-based multi-objective optimization, it provides an efficient feedback of environmental performance in the early design stages. It reduces the time consumption for environmental performance optimization from hours (or even days if the scale is larger) to seconds.

We designed three encoding methods to transform the simulated data into a useable format for GAN and compared the predictive performance of pix2pix, cycleGAN, and ANN for the three environmental indicators. On the test set, the pix2pix model using the “clone” encoding method predicts PLW, radiation and UTCI with R^2 of 0.70, 0.86 and

0.80. The result shows that the prediction accuracy of the “clone” and “3in1” methods are close. However, the “3in1” coding method has more potential in the future because it has the advantages of short training time and small model size. To increase the generalizability of the proposed method, we used the unaffected reference environmental data of the site as the basis for normalization. It allows the trained model to be applied to regions with varying meteorological conditions.

We compared three downsampling methods to transform the output of the GAN into an optimization objective. Six schemes were designed to compare different downsampling methods, different optimization algorithms, and different surrogate models. The results show that the NSGA-II combined with the GAP downsampling method obtains the best optimization results. The comparison of GAN-based and ANN-based optimization schemes demonstrate the importance of obtaining global environmental information in environment optimization of urban block.

In this study, GANs can speed up the environmental performance evaluation by 120–240 times. Considering the dataset construction and deep model training and hyperparameter optimization, we suggest using GAN-based surrogate models for multi-objective optimization when the number of optimized samples exceeds 174.

This study promotes training deep generative models, especially GANs in outdoor environment assessment and optimization. However, the prediction performance of the current GAN is near the upper limit, so increasing the amount of data and hyperparametric optimization can hardly improve the model performance further. More accurate prediction of PLW and UTCI spatial distributions is currently a challenge. It will greatly affect the performance of multi-objective optimization. The accuracy bound of purely data-driven methods in predicting physical problems may be broken by introducing physical rules. Physically informed neural networks combined with deep generative models may be a way to achieve more accurate PLW and UTCI predictions in the future.

CRediT authorship contribution statement

Chenyu Huang: Writing – original draft, Software, Methodology, Formal analysis, Conceptualization. **Gengjia Zhang:** Visualization, Software, Resources. **Jiawei Yao:** Writing – review & editing, Validation, Supervision, Project administration, Formal analysis, Conceptualization. **Xiaoxin Wang:** Software, Formal analysis. **John Kaiser Calautit:** Writing – review & editing. **Cairong Zhao:** Writing – review & editing, Supervision. **Na An:** Methodology. **Xi Peng:** Software.

Declaration of competing interest

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

Data availability

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

Acknowledgements

This research was financially supported by the National Natural Science Foundation of China under Grant NO. 51908410; the Shanghai Municipal Science and Technology Major Project under Grant NO. 2021SHZDZX0100 Fundamental Research Funds for the Central Universities. Moreover, the authors would like to thank Ruizhe Luo and Jinyu Wang for editing the figures.

Appendix 1. Wind tunnel data validation and grid sensitivity analysis

The CFD model was validated by the experimental measurements reported by the AIJ for the Case C wind tunnel [64]. The prototype contains nine cubes, each measuring $0.2 \text{ m} \times 0.2 \text{ m} \times 0.2 \text{ m}$ (length \times width \times height). All street canyons between the two cubes are planned with the same uniform aspect ratio. 120 monitoring points were placed within the canyon at 0.02 m from the ground. We used the same computational domain settings described in the Method section and the logarithmic wind profile provided by the wind tunnel data for the calculations. The results are presented in Fig. A1.

Three structured grids were constructed. The coarse grid has a minimum grid size of 0.01 m containing 1876160 cells, the base grid has a minimum size of 0.005 m containing 4172190 cells, and the fine grid has a minimum grid size of 0.0025 m containing 5574240 cells (Fig. A1(a)). As in the wind tunnel experiments setup, we simulated the case of a west wind (W). The wind velocity ratio was defined by dividing the 3D wind speed U at the monitoring point by the reference wind speed U_0 provided by wind tunnel experiments. We perform a simple linear fit to the data of monitoring points for different grid sizes. The different grid sizes are compared in Fig. A1(c) and (d), where the difference between the coarse grid and the basic grid is significantly larger than the difference between the basic grid and the fine grid. It indicates that the basic grid is a good compromise between computational accuracy and computational cost. Our grid setting for the parametric urban block is expanded to 1:100 based on the minimum size of the basic grid.

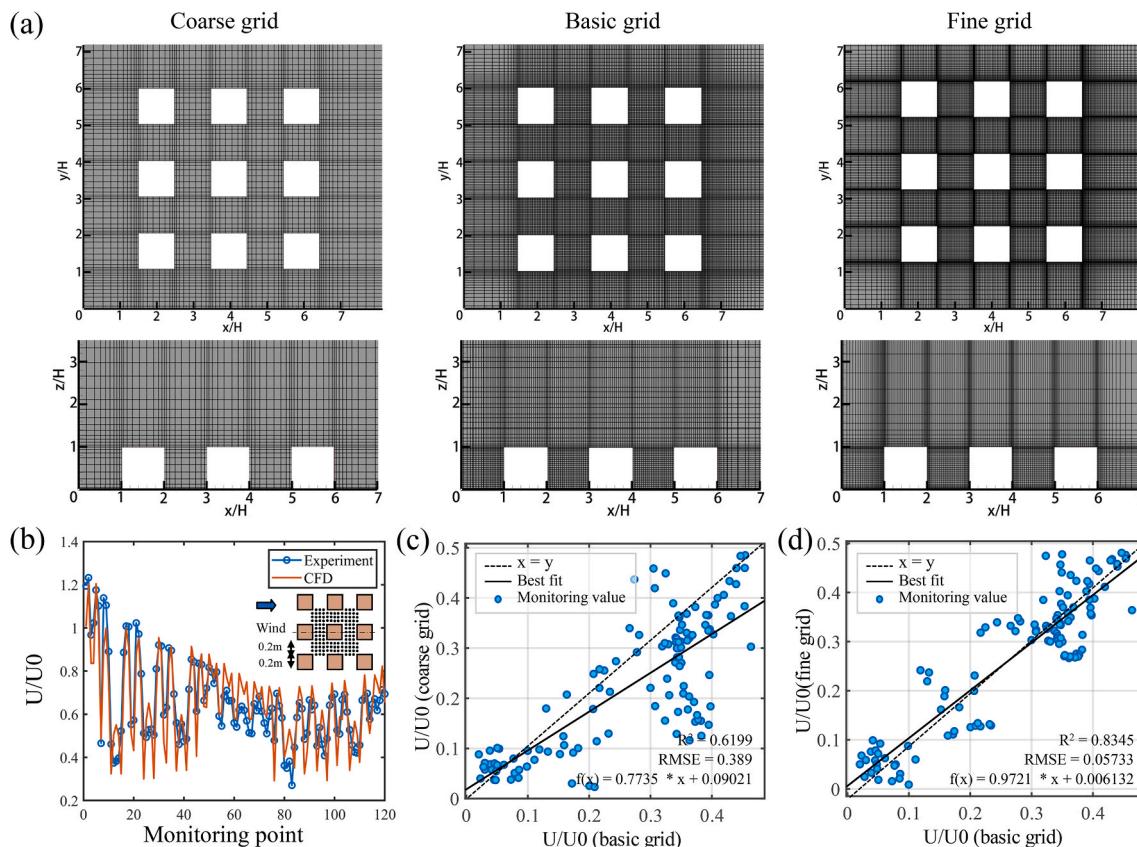


Fig. A1. Wind tunnel data validation and grid sensitivity analysis (a) Structured grid in three sizes, (b) Comparison of velocity ratio of wind tunnel experiments with CFD at monitoring points, (c) Comparison of velocity ratio for the coarse and essential grid at monitoring points, (d) same for the primary and fine grid.

Appendix 2. Data augmentation in GAN training

We have adjusted the way the data are augmented in the source code. Because the original authors proposed pix2pix when learning natural images, the convolutional layers' rotation, translation, and scale invariance were enhanced using scaling, rotation, and central symmetry in the data augmentation process. When predicting the PLW, considering rotation invariance means that the wind direction would become unfixed in the training process. Fig. A2 shows the visualization of the input data using the resampling, random cropping method during data augmentation versus random mirroring on top of the resampling and random cropping. Fig. A3 compares the model prediction results of different data augmentation methods.

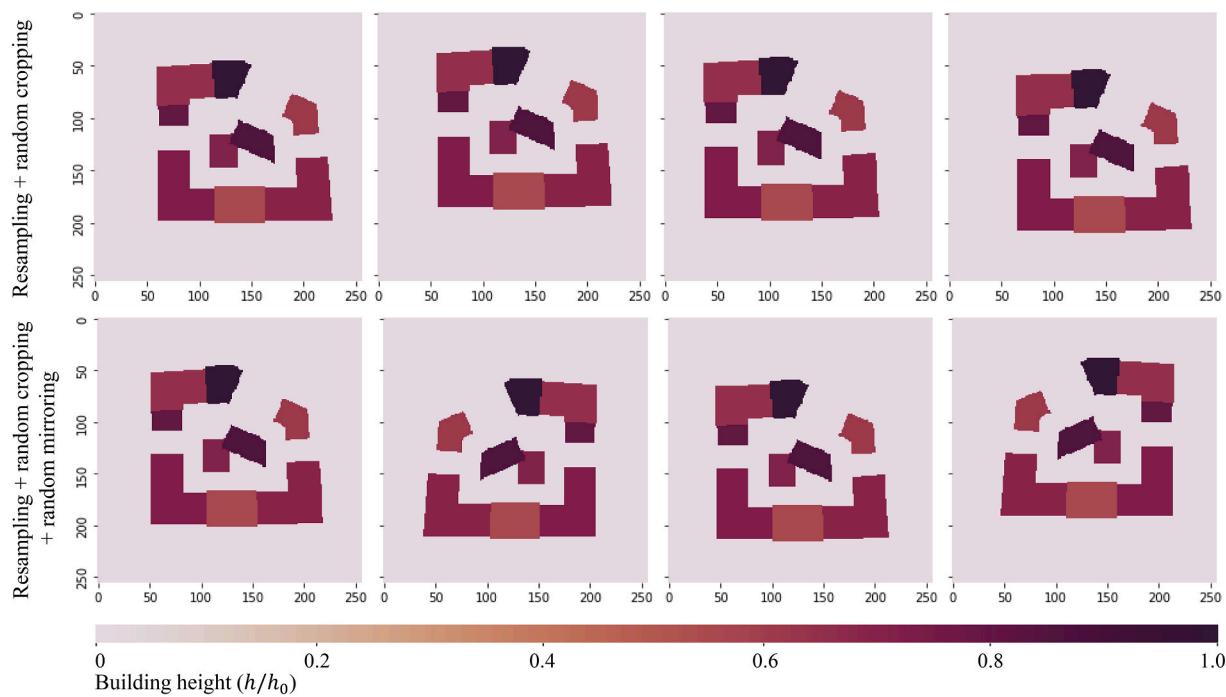


Fig. A2. Comparison of data augmentation methods.

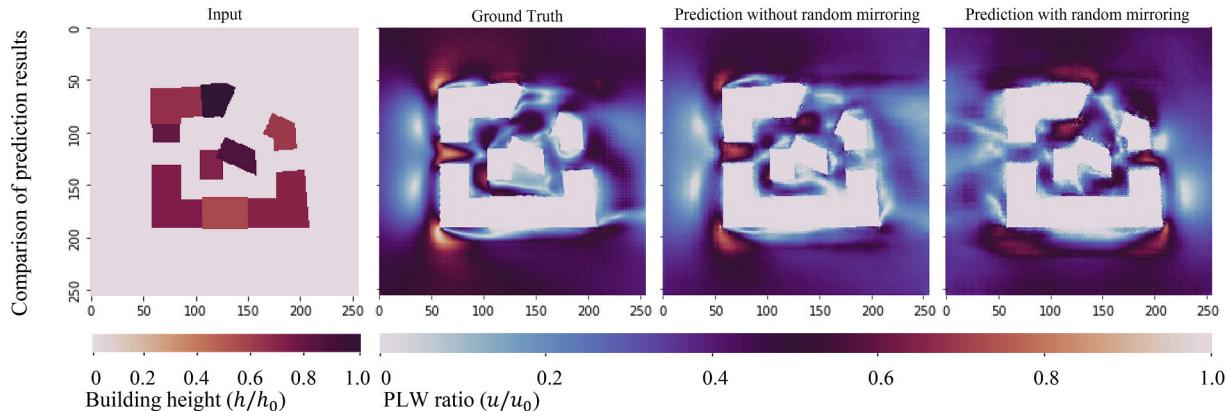


Fig. A3. Results of different data augmentation methods.

References

- [1] V.J. Lee, M. Ho, C.W. Kai, X. Aguilera, D. Heymann, A. Wilder-Smith, Epidemic preparedness in urban settings: new challenges and opportunities, *Lancet Infect. Dis.* 20 (2020) 527–529, [https://doi.org/10.1016/S1473-3099\(20\)30249-8](https://doi.org/10.1016/S1473-3099(20)30249-8).
- [2] H. Gu, Y. Cao, E. Elahi, S.K. Jha, Human health damages related to air pollution in China, *Environ. Sci. Pollut. Res. Int.* 26 (2019) 13115–13125, <https://doi.org/10.1007/s11356-019-04708-y>.
- [3] N. Singh, S. Singh, R. Mall, Urban ecology and human health: implications of urban heat island, air pollution and climate change nexus, *Urban Ecol.* (2020), <https://doi.org/10.1016/b978-0-12-820730-7.00017-3>.
- [4] E. Ng, Policies and technical guidelines for urban planning of high-density cities – air ventilation assessment (AVA) of Hong Kong, *Build. Environ.* 44 (2009) 1478–1488, <https://doi.org/10.1016/j.buildenv.2008.06.013>.
- [5] A. Sharifi, A.R. Khavarian-Garmsir, The COVID-19 pandemic: impacts on cities and major lessons for urban planning, design, and management, *Sci. Total Environ.* 749 (2020), 142391, <https://doi.org/10.1016/j.scitotenv.2020.142391>.
- [6] F. Yang, L. Chen, Developing a thermal atlas for climate-responsive urban design based on empirical modeling and urban morphological analysis, *Energy Build.* 111 (2016) 120–130, <https://doi.org/10.1016/j.enbuild.2015.11.047>.
- [7] A. Dawodu, A. Cheshmehzangi, Impact of floor area ratio (FAR) on energy consumption at meso scale in China: case study of ningbo, *Energy Proc.* 105 (2017) 3449–3455, <https://doi.org/10.1016/j.egypro.2017.03.789>.
- [8] Z. Li, H. Zhang, C.-Y. Wen, A.-S. Yang, Y.-H. Juan, Effects of frontal area density on outdoor thermal comfort and air quality, *Build. Environ.* 180 (2020), 107028, <https://doi.org/10.1016/j.buildenv.2020.107028>.
- [9] Y. Shi, X. Xie, J.C.-H. Fung, E. Ng, Identifying critical building morphological design factors of street-level air pollution dispersion in high-density built environment using mobile monitoring, *Build. Environ.* 128 (2018) 248–259, <https://doi.org/10.1016/j.buildenv.2017.11.043>.
- [10] T. Ma, T. Chen, Classification and pedestrian-level wind environment assessment among Tianjin's residential area based on numerical simulation, *Urban Clim.* 34 (2020), 100702, <https://doi.org/10.1016/j.uclim.2020.100702>.
- [11] S. Park, S.E. Tuller, M. Jo, Application of Universal Thermal Climate Index (UTCI) for microclimatic analysis in urban thermal environments, *Landsc. Urban Plann.* 125 (2014) 146–155, <https://doi.org/10.1016/j.landurbplan.2014.02.014>.
- [12] P. Rode, C. Keim, G. Robazza, P. Viejo, J. Schofield, Cities and energy: urban morphology and residential heat-energy demand, *Environ. Plann. B* 41 (2014) 138–162, <https://doi.org/10.1088/b39065>.
- [13] H. Takebayashi, E. Ishii, M. Moriyama, A. Sakaki, S. Nakajima, H. Ueda, Study to examine the potential for solar energy utilization based on the relationship between urban morphology and solar radiation gain on building rooftops and wall surfaces, *Sol. Energy* 119 (2015) 362–369, <https://doi.org/10.1016/j.solener.2015.05.039>.
- [14] H. Leng, X. Chen, Y. Ma, N.H. Wong, T. Ming, Urban morphology and building heating energy consumption: evidence from Harbin, a severe cold region city, *Energy Build.* 224 (2020), 110143, <https://doi.org/10.1016/j.enbuild.2020.110143>.

- [15] R. Privitera, V. Palermo, F. Martinico, A. Fichera, D. La Rosa, Towards lower carbon cities: urban morphology contribution in climate change adaptation strategies, *Eur. Plann. Stud.* 26 (2018) 812–837, <https://doi.org/10.1080/09654313.2018.1426735>.
- [16] X. Shi, Design optimization of insulation usage and space conditioning load using energy simulation and genetic algorithm, *Energy* 36 (2011) 1659–1667.
- [17] P.F. Yuan, Y. Song, Y. Lin, H.S. Beh, Y. Chao, T. Xiao, S. Huang, J. Zheng, Z. Wu, An architectural building cluster morphology generation method to perceive, derive, and form based on cyborg-physical wind tunnel (CPWT), *Build. Environ.* 203 (2021), 108045, <https://doi.org/10.1016/j.buildenv.2021.108045>.
- [18] Z. Li, H. Chen, B. Lin, Y. Zhu, Fast bidirectional building performance optimization at the early design stage, *Build. Simulat.* 11 (2018) 647–661, <https://doi.org/10.1007/s12273-018-0432-1>.
- [19] P.B. Purup, S. Petersen, Research framework for development of building performance simulation tools for early design stages, *Autom. ConStruct.* 109 (2020), 102966, <https://doi.org/10.1016/j.autcon.2019.102966>.
- [20] P. Kastner, T. Dogan, Eddy3D: a toolkit for decoupled outdoor thermal comfort simulations in urban areas, *Build. Environ.* 212 (2022), 108639, <https://doi.org/10.1016/j.buildenv.2021.108639>.
- [21] M.P. Mostapha Sadeghipour Roudsari, LADYBUG: A Parametric Environmental Plugin for Grasshopper to Help Designers Create an Environmentally-Conscious Design, AIVC, 2014. <https://www.aivc.org/resource/ladybug-parametric-environmental-plugin-grasshopper-help-designers-create-environmentally>. (Accessed 8 August 2022).
- [22] M. Mortezaeza, L. Leon, Wang, Solving city and building microclimates by fast fluid dynamics with large timesteps and coarse meshes, *Build. Environ.* 179 (2020), 106955, <https://doi.org/10.1016/j.buildenv.2020.106955>.
- [23] S. Zheng, Z.J. Zhai, Y. Wang, Y. Xue, L. Duanmu, W. Liu, Evaluation and comparison of various fast fluid dynamics modeling methods for predicting airflow around buildings, *Build. Simulat.* 15 (2022) 1083–1095, <https://doi.org/10.1007/s12273-021-0860-1>.
- [24] B. Bozorgmehr, P. Willemesen, J.A. Gibbs, R. Stoll, J.-J. Kim, E.R. Pardyjak, Utilizing dynamic parallelism in CUDA to accelerate a 3D red-black successive over relaxation wind-field solver, *Environ. Model. Software* 137 (2021), 104958, <https://doi.org/10.1016/j.envsoft.2021.104958>.
- [25] M. Overby, P. Willemesen, B.N. Bailey, S. Halverson, E.R. Pardyjak, A rapid and scalable radiation transfer model for complex urban domains, *Urban Clim.* 15 (2016) 25–44, <https://doi.org/10.1016/j.ulclim.2015.11.004>.
- [26] F. Bre, N. Roman, V.D. Fachinotti, An efficient metamodel-based method to carry out multi-objective building performance optimizations, *Energy Build.* 206 (2020), 109576, <https://doi.org/10.1016/j.enbuild.2019.109576>.
- [27] S. Attia, M. Hamdy, W. O'Brien, S. Carlucci, Assessing gaps and needs for integrating building performance optimization tools in net zero energy buildings design, *Energy Build.* 60 (2013) 110–124, <https://doi.org/10.1016/j.enbuild.2013.01.016>.
- [28] E. Toulopaki, T. Theodosiou, Performance simulation integrated in parametric 3D modeling as a method for early stage design optimization—a review, *Energies* 10 (2017) 637, <https://doi.org/10.3390/en10050637>.
- [29] B. Ekici, C. Cubukcuoglu, M. Turrin, I.S. Sarıyıldız, Performative computational architecture using swarm and evolutionary optimisation: a review, *Build. Environ.* 147 (2019) 356–371, <https://doi.org/10.1016/j.buildenv.2018.10.023>.
- [30] A.A.S. Bahdad, S.F.S. Fadzil, H.O. Onubi, S.A. BenLasod, Sensitivity analysis linked to multi-objective optimization for adjustments of light-shelves design parameters in response to visual comfort and thermal energy performance, *J. Build. Eng.* 44 (2021), 102996, <https://doi.org/10.1016/j.jobe.2021.102996>.
- [31] N. Nasrollahzadeh, Comprehensive building envelope optimization: improving energy, daylight, and thermal comfort performance of the dwelling unit, *J. Build. Eng.* 44 (2021), 103418, <https://doi.org/10.1016/j.jobe.2021.103418>.
- [32] J. Natanian, T. Wortmann, Simplified evaluation metrics for generative energy-driven urban design: a morphological study of residential blocks in Tel Aviv, *Energy Build.* 240 (2021), 110916, <https://doi.org/10.1016/j.enbuild.2021.110916>.
- [33] S. Mirzabeigi, M. Razkenari, Design optimization of urban typologies: a framework for evaluating building energy performance and outdoor thermal comfort, *Sustain. Cities Soc.* 76 (2022), 103515, <https://doi.org/10.1016/j.scs.2021.103515>.
- [34] S. Chang, N. Saha, D. Castro-Lacouture, P.P.-J. Yang, Multivariate relationships between campus design parameters and energy performance using reinforcement learning and parametric modeling, *Appl. Energy* 249 (2019) 253–264, <https://doi.org/10.1016/j.apenergy.2019.04.109>.
- [35] S. Liu, W. Pan, H. Zhang, X. Cheng, Z. Long, Q. Chen, CFD simulations of wind distribution in an urban community with a full-scale geometrical model, *Build. Environ.* 117 (2017) 11–23, <https://doi.org/10.1016/j.buildenv.2017.02.021>.
- [36] B. Lin, H. Chen, Q. Yu, X. Zhou, S. Lv, Q. He, Z. Li, MOOSAS—A systematic solution for multiple objective building performance optimization in the early design stage, *Build. Environ.* 200 (2021) 107929.
- [37] S. Seyedzadeh, F.P. Rahimian, I. Glesk, M. Roper, Machine learning for estimation of building energy consumption and performance: a review, *Visualiz. Eng.* 6 (2018) 5, <https://doi.org/10.1186/s40327-018-0064-7>.
- [38] A. Nutkiewicz, Z. Yang, R.K. Jain, Data-driven Urban Energy Simulation (DUE-S): a framework for integrating engineering simulation and machine learning methods in a multi-scale urban energy modeling workflow, *Appl. Energy* 225 (2018) 1176–1189, <https://doi.org/10.1016/j.apenergy.2018.05.023>.
- [39] H. Sha, M. Moujahed, D. Qi, Machine learning-based cooling load prediction and optimal control for mechanical ventilative cooling in high-rise buildings, *Energy Build.* 242 (2021), 110980, <https://doi.org/10.1016/j.enbuild.2021.110980>.
- [40] A.U. Weerasuriya, X. Zhang, B. Lu, K.T. Tse, C.-H. Liu, Optimizing lift-up design to maximize pedestrian wind and thermal comfort in ‘hot-calm’ and ‘cold-windy’ climates, *Sustain. Cities Soc.* 58 (2020), 102146, <https://doi.org/10.1016/j.scs.2020.102146>.
- [41] Y. Han, L. Shen, C. Sun, Developing a parametric morphable annual daylight prediction model with improved generalization capability for the early stages of office building design, *Build. Environ.* 200 (2021), 107932, <https://doi.org/10.1016/j.buildenv.2021.107932>.
- [42] S. Wang, Y.K. Yi, N. Liu, Multi-objective optimization (MOO) for high-rise residential buildings’ layout centered on daylight, visual, and outdoor thermal metrics in China, *Build. Environ.* 205 (2021), 108263, <https://doi.org/10.1016/j.buildenv.2021.108263>.
- [43] W. Wang, K. Liu, M. Zhang, Y. Shen, R. Jing, X. Xu, From simulation to data-driven approach: a framework of integrating urban morphology to low-energy urban design, *Renew. Energy* 179 (2021), <https://doi.org/10.1016/j.renene.2021.08.024>, 2016–2035.
- [44] Y. Wu, Q. Zhan, S.J. Quan, Y. Fan, Y. Yang, A surrogate-assisted optimization framework for microclimate-sensitive urban design practice, *Build. Environ.* 195 (2021), 107661, <https://doi.org/10.1016/j.buildenv.2021.107661>.
- [45] G. Calzolari, W. Liu, Deep learning to replace, improve, or aid CFD analysis in built environment applications: a review, *Build. Environ.* 206 (2021), 108315, <https://doi.org/10.1016/j.buildenv.2021.108315>.
- [46] D. Kochkov, J.A. Smith, A. Alieva, Q. Wang, M.P. Brenner, S. Hoyer, Machine learning-accelerated computational fluid dynamics, *Proc. Natl. Acad. Sci. U. S. A.* 118 (2021), e2101784118, <https://doi.org/10.1073/pnas.2101784118>.
- [47] B. Kim, D.-E. Lee, K.R.S. Preethaa, G. Hu, Y. Natarajan, K.C.S. Kwok, Predicting wind flow around buildings using deep learning, *J. Wind Eng. Ind. Aerod.* 219 (2021), 104820, <https://doi.org/10.1016/j.jweia.2021.104820>.
- [48] H. Tanaka, Y. Matsuoka, T. Kawakami, Y. Azegami, M. Yamamoto, K. Ohtake, T. Some, Optimization calculations and machine learning aimed at reduction of wind forces acting on tall buildings and mitigation of wind environment, *Int. J. High-Rise Build.* 8 (2019) 291–302, <https://doi.org/10.21022/IJHRB.2019.8.4.291>.
- [49] J. Musil, J. Knir, A. Vitsas, I. Gallou, Towards sustainable architecture: 3D convolutional neural networks for computational fluid dynamics simulation and reverse DesignWorkflow, <https://doi.org/10.48550/arXiv.1912.02125>, 2019.
- [50] S. Ravuri, K. Lenc, M. Willson, D. Kangin, R. Lam, P. Mirowski, M. Fitzsimons, M. Athanasiadou, S. Kashem, S. Madge, R. Prudden, A. Mandhane, A. Clark, A. Brock, K. Simonyan, R. Hadsell, N. Robinson, E. Clancy, A. Arribas, S. Mohamed, Skilful precipitation nowcasting using deep generative models of radar, *Nature* 597 (2021) 672–677, <https://doi.org/10.1038/s41586-021-03854-z>.
- [51] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-Image translation with conditional adversarial networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5967–5976, <https://doi.org/10.1109/CVPR.2017.632>.
- [52] Q. He, Z. Li, W. Gao, H. Chen, X. Wu, X. Cheng, B. Lin, Predictive models for daylight performance of general floorplans based on CNN and GAN: a proof-of-concept study, *Build. Environ.* 206 (2021), 108346, <https://doi.org/10.1016/j.buildenv.2021.108346>.
- [53] S. Mokhtar, A. Sojka, C.C. Davila, Conditional generative adversarial networks for pedestrian wind flow approximation, in: Proceedings of the 11th Annual Symposium on Simulation for Architecture and Urban Design, Society for Computer Simulation International, San Diego, CA, USA, 2020, pp. 1–8.
- [54] S. Duering, A. Chronic, R. Koenig, Optimizing urban systems: integrated optimization of spatial configurations, in: Proceedings of the 11th Annual Symposium on Simulation for Architecture and Urban Design, Society for Computer Simulation International, San Diego, CA, USA, 2020, pp. 1–7.
- [55] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2242–2251, <https://doi.org/10.1109/ICCV.2017.244>.
- [56] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, B. Catanzaro, High-resolution image synthesis and semantic manipulation with conditional GANs. <https://doi.org/10.48550/arXiv.1711.11585>, 2018.
- [57] J. Tian, S. Xu, A morphology-based evaluation on block-scale solar potential for residential area in central China, *Sol. Energy* 221 (2021) 332–347, <https://doi.org/10.1016/j.solener.2021.02.049>.
- [58] H. Yu, M. Wang, X. Lin, H. Guo, H. Liu, Y. Zhao, H. Wang, C. Li, R. Jing, Prioritizing urban planning factors on community energy performance based on GIS-informed building energy modeling, *Energy Build.* 249 (2021), 111191, <https://doi.org/10.1016/j.enbuild.2021.111191>.
- [59] J. Liu, J. Niu, Y. Du, C.M. Mak, Y. Zhang, LES for pedestrian level wind around an idealized building array—assessment of sensitivity to influencing parameters, *Sustain. Cities Soc.* 44 (2019) 406–415, <https://doi.org/10.1016/j.scs.2018.10.034>.
- [60] P. Bröde, E.L. Krüger, F.A. Rossi, D. Fiala, Predicting urban outdoor thermal comfort by the Universal Thermal Climate Index UTCI—a case study in Southern Brazil, *Int. J. Biometeorol.* 56 (2012) 471–480, <https://doi.org/10.1007/s00484-011-0452-3>.
- [61] K. Blazejczyk, Y. Epstein, G. Jendritzky, H. Staiger, B. Tinz, Comparison of UTCI to selected thermal indices, *Int. J. Biometeorol.* 56 (2012) 515–535, <https://doi.org/10.1007/s00484-011-0453-2>.
- [62] J. Franke, A. Hellsten, K. Schlünzen, B. Carissimo, Best practice guideline for the CFD simulation of flows in the urban environment—a summary, in: Paper presented

- at the 11th Conference on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes, Cambridge, UK, July 2007, 2007.
- [63] Y. Tominaga, A. Mochida, R. Yoshie, H. Kataoka, T. Nozu, M. Yoshikawa, T. Shirasawa, AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings, *J. Wind Eng. Ind. Aerodynam.* 96 (10–11) (2008) 1749–1761.
- [64] Y. Nonomura, N. Kobayashi, Y. Tominaga, A. Mochida, The cross comparison of CFD results for flow field around building models (Part 3)-Wind tunnel test for the verification of models for the flow field around building blocks, in: Paper presented at the Summaries of Technical Papers of Annual Meeting, Japan Association for Wind Engineering, 2003.