A NOVEL GAN-BASED METHOD FOR BUILDING SURFACE WIND PRESSURE PREDICTION

Lin Sun
Shuqi Cao
Likai Wang
Guohua Ji

School of Architecture and Urban Planning
Nanjing University
22 Hankou Road, Nanjing, CHINA
dg1736001@smail.nju.edu.cn,
dg20360001@smail.nju.edu.cn,
wang.likai@outlook.com, jgh@nju.edu.cn

ABSTRACT
The wind pressure on the surface of buildings can be critical, especially for high-rise buildings. To overcome the limitations in applying time-consuming CFD simulations to early design stage, we introduce a novel approach for generating real-time surface wind pressure in high-rise buildings through Deep Learning (DL). The DL model is trained on datasets of hundreds of buildings geometries generated by Grasshopper plugin EvoMass, and their corresponding surface wind pressure is simulated by RhinoCFD. Moreover, regarding the current performance prediction using DL, we have noted that the common labeling map and DL model may hinder the efficiency of training. Thus, we proposed a new labeling approach and a modified DL model to assess the accuracy of the prediction. To demonstrate the validity of the approach, the experiments of simulation predictions were conducted, and the results shows that the presented method can greatly enhance the accuracy.

Keywords: Deep Learning, Surface Wind Pressure Simulation, Surrogate Model.

1 INTRODUCTION
As contemporary cities are built increasingly denser and higher, high-rise buildings can be subjected to extreme wind pressure loads as wind velocity increases over altitude. The most common approach employed to predict wind loads on buildings is numerical simulation. The use of numerical simulation to calculate the wind load of buildings can evaluate various wind related factors, which greatly promotes the development of the research on building wind resistance (Ostergard et al. 2016).

Although computer simulation technology has made great progress, the numerical simulation typically lasts for several hours due to a large amount of computation, and a design optimization process include thousands of times design generation and the corresponding simulation analysis. The time, cost and computational expense of simulation made design workflow ineffective. Therefore, it is not practical for architects to conduct a large amount of performance analysis and design adjustment in the early design stage for performance-driven architectural design (Attia et al. 2013).
In order to increase the efficiency of simulations in early design optimization workflows, this paper proposes a Deep Learning framework on the Grasshopper platform to offer a real-time prediction of surface wind pressure in high-rise buildings, where no simulation is required. The DL framework is trained on datasets of hundreds of high-rise building geometries generated by Grasshopper plugin EvoMass, and their corresponding surface wind pressure simulated by RhinoCFD.

This research is part of a broader project center around developing a system that incorporates DL models to handle different building shape features and predict building environmental performance evaluation. As an initial phase, this paper focuses on applying DL model to predict surface wind pressure in high-rise buildings. To verify the efficacy of the proposed method, the experiments of simulation predictions were conducted. Moreover, a new labeling approach and network architecture, which can increase the accuracy of the prediction was developed as an alternative of the common method.

The main contribution of this research is the exploration of the suitability of this method for the prediction of surface wind pressure in high-rise buildings in early design stages, and the evaluation of different training set encoding strategies to identify what simplifications are acceptable for this purpose.

2 RELATED WORK

Computational Fluid Dynamics (CFD) tools have been widely applied for simulating fluid flow and heat transfer numerically. Many different numerical methods have also been developed by researchers in the past decades using this tool to simulate a wide range of complex flows and heat transfer conditions. However, the heavy computation and time cost dramatically limits its applicating use for controlling purposes.

In order to increase the calculation speed of CFD simulations in early design optimization workflows (Chronis et al. 2017), various approaches have been developed. The Dexen framework developed by Janssen uses parallel computing to disperse the performance simulation analysis of design individuals in the optimization process to multiple computers for parallel calculation to increase the speed of the optimization process. Kyropoulou applies the computing power of cloud servers to implement "cloud computing" for building performance simulation (Kyropoulou et al. 2018). Furthermore, the Procedural plugin on Rhino-Grasshopper platform also provides cloud computing functions. The above approaches have achieved faster building performance simulation by enhancing computing resources. However, due to the complexity of the technology or the high cost of cloud computing, such approaches have not been widely adopted in practice.

In recent years, Deep Learning approaches have been proposed as an alternative to performance-driven design, since DL models can process, learn, and synthesize data, it can assist designers in effective decision making by enabling prediction of environmental metrics of their designs (Spacemaker et al. 2020). DL works on the principle of identifying patterns in observed data and thus imparting experiential knowledge upon computer systems and therefore gaining insights from the data (Herbon 2016). The outcomes of the simulation show that an DL algorithm can be trained to detect the patterns produced by the environmental analytical data and thus perform predictions that are usable in early design processes (Sebestyen and Tyc 2020). Since no actual physical simulations are required, this type of approach can provide nearly "real-time" feedback on building performance.

In recent research, some researchers have applied DL models to performance-driven design, such as Grasshopper plugins Ant and Dodo, which can predict simulation data through methods such as linear regression, and support vector machines. Although these plugins allow users to apply artificial neural networks (ANN) to predict simulations, they are limited in geometrical parameters and do not provide full field wind flow data for the spaces around buildings and not suitable for common design workflows because of the complicated systems. The NumPy and Pandas libraries for Python are more suitable for these tasks, but they do not run natively inside Grasshopper's Gh-Python environment. Instead the Tensorflow library was used inside Python-Anaconda for building and training DL networks. The Tensorflow library gave
precise control over the exact neural network architecture and allowed to utilize the computer's GPU to perform very fast DL training (Géron 2019). Based on the Tensorflow library, performance-based DL projects have been proposed. Sarah presented a conditional generative adversarial network approach to developing a surrogate model for the inference of the design contribution of urban morphologies to pedestrian wind flow conditions for any given urban configuration (Mokhtar et al. 2020).

Although DL models can only provide approximate simulations, it is suitable for the early design stages. This is due to the designers need to evaluate multiple design solutions swiftly, the efficiency is more significant in the early phase.

3 METHOD

This paper employs a DL approach that integrates neural network and CFD tool to achieve a real-time prediction of surface wind pressure in high-rise building models based only on the input, demanding no simulation. Moreover, in this paper, a new labeling approach was developed instead of the common method which uses different colors to convert a 3D geometry representation into a 2D image. In this research, the workflow included 4 procedures (Figure 1).

- Setting random parameters to generate the building geometries in Grasshopper.
- Inventing labeling map through the elevation of the building geometries.
- The surface wind pressure simulation on the generated models was conducted.
- The outcomes of the simulation were imported to the DL framework for the training process.

![Figure 1: The overview of the workflow](image-url)
3.1 Building Geometry Generation

The purpose of the generative part is to create DL models based on different complexities of training data for comparison. This procedure was based on an additive form generative algorithm, within Rhino-Grasshopper plugin EvoMass (Wang et al. 2020). This algorithm creates building massing by accumulating several mass elements, allowing the generated design to have high variability (Figure 2). The generated building massing design variants can be controlled by several user-defined parameters, such as plane size, floor number, floor height etc. Once the settings are defined, the generated design will be used as the basis for the following procedure. To produce large datasets for training, randomized massing was created. In this process, 1000 different massings were created randomly.

![Figure 2. Samples of building geometries generation](image)

3.2 Labeling Maps

For training a supervised learning DL model, it is required to feed the algorithm with both input data and the desired output solutions corresponding to these inputs. In this study, the data of input and output were both created in Grasshopper. After the building geometries were generated, the 3D geometry needs to be converted to a 2D image. Next, we created a labeling rule which uses different colors to represent the depth of each part of the building mass in elevation.

Colors with Gray-Scale values were commonly used in the labeling map in order to convert a 3D geometry representation into a 2D image. The light gray color represents the low depth and the dark gray color for the high depth.
However, we observed that in the performance simulation of facades, the wind pressure typically changes more dramatically beside the edges of the building surface. On account of this phenomenon, we hypothesize the shadow map may be more effective representing wind pressure in this research. Therefore, we use the building shadow map generated by the “Rendering Mode” of Rhino as the labeling map in contrast with the Gray-scale map. Based on these two labeling rules, two labeled datasets were created for comparison. Figure 3 and Figure 4 showed the two different labeling maps respectively.

![Figure 3. Sample of elevations in Gray-scale map](image1)

![Figure 4. Sample of elevations in shadow map](image2)

### 3.3 Surface Wind Pressure Simulation

The simulation is conducted to calculate the surface wind pressure of the generated models. In this study, the simulation is executed by RhinoCFD, a Computational Fluid Dynamics software plugin, built directly into the Rhino environment and powered by CHAM’s PHOENICS. It allows users to simulate the interaction of their models with the surrounding fluid, thus enabling rapid optimization and testing without leaving the familiarity of the Rhino environment.

The RhinoCFD solver was used to solve the governing equations of steady-state Reynolds Averaging Navier-Stokes (RANS) with a turbulence model of realizable k-ε, following the support in the literature and industry standards for the robustness of this CFD approach for pedestrian wind level (PLW) applications.
The computing domain grid was defined as 6 times the maximum building height on the windward and sides, 3 times on the height, and 15 times on the leeward side. In the mesh setting, we set 50x50x50 as the number of cells, and 1m as the minimum cell size. A flat terrain was assumed for all cases and a landscape roughness of a dense urban environment was used. At the upstream inlet, a reference wind speed of 3.5 m/s at a 10m reference height was used for all simulations.

Figure 5 shows sample of simulations calculated by RhinoCFD. After simulating, 1000 pairs of elevation-wind pressure load simulation images in each labeled dataset as mentioned in 3.2, to be used as the training dataset.

3.4 The Deep Learning Model Training

In order to build the DL model, the Keras was employed with the Tensorflow library as the backend. For employing native Python library, the Grasshopper plugin GH_CPython was used. This plugin provides a component that implements CPython codes inside grasshopper which can allow users to employ most of the Python libraries.

With the training dataset, the Pix2Pix model (Isola et al. 2017) is commonly used. The pix2pix is a conditional Generative Adversarial Network (cGAN) framework for image-to-image translation, such as generating a real photo from a partly-damaged photo, a colorful map from a black-and-white map, and an image with texture and shadow from a linear sketch.
It consists of a generator G, which translates semantic label maps to realistic-looking images; and a discriminator D, which distinguishes the fake images. The advantage of the Pix2Pix model is that it uses the GAN framework to provide a general framework for issues of "Image-to-Image translation". However, we found that in the related research of performance simulation, the functionality of the discriminator D in the Pix2Pix framework will be greatly reduced, and the training efficiency will be dragged down. Based on the Pix2Pix framework, we developed a new network architecture named “Pix-G”. In this framework, we removed the discriminator D and modified the framework of generator G (Figure 6). In this research, we used Tensorflow 2.0 as the backend for building the training model, and the version of Python Anaconda was 3.7.2.

The input and output were set to 256x256 pixels and using kaiming weight initialization, batch size of 32, initial learning rate of 0.0002. After every 50 epochs, the trained weights were saved in order to monitor the progress of the training. In epoch 800, inaccuracies still occurred occasionally, but the output images presented the changing pattern from input images to the ground truth images. Thus, we decided to stop training at epoch 800. And after training, we can input an image and tell the program to generate the most possible corresponding output image. The training process was completed on a computer with a GeForce RTX 3070 Graphics Card, and each epoch took 20 seconds on average.

4 RESULT & VALIDATION

Comparing the predictions produced from the DL model and the simulations from RhinoCFD confirmed that the DL model was able to predict surface wind pressure successfully. To evaluate the accuracy of the predictions, the “Structural Similarity Index Measure (SSIM)” value was calculated for each facade prediction to show which combination of the labeling map and network architecture performed the best and worse. SSIM is a method for image quality assessment that involves extracting structural information and measuring the degradation of the structural data for the images under analysis. The closer the value of SSIM is to 1, the higher the similarity between the two images.

Based on the validation datasets containing 100 paired images, the predictions of wind pressure simulation were compared against the RhinoCFD simulation. Figure 7 showcases the average similarity value and overall similarity numerical range calculated by SSIM. As shown in Figure 6, it is clear that the shadow map coupled with the Pix-G framework achieved the best scores.
We chose one sample from each facade which used different labeling maps and trained by Pix2Pix/Pix-G model to show the assessment results of the SSIM in Figure 8 and Figure 9, and each sample contains input labeling map, prediction and the true images. As the showing of these 2 figures, the colours in outputs produced by shadow map have clearly distributed especially in the edges of surface. In addition, it is clear to see that the predicted images generated by Pix-G have a higher accuracy.

<table>
<thead>
<tr>
<th>Combination</th>
<th>SSIM</th>
<th>Front</th>
<th>Right</th>
<th>Left</th>
<th>Back</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average</td>
<td>0.924</td>
<td>0.912</td>
<td>0.916</td>
<td>0.904</td>
</tr>
<tr>
<td></td>
<td>Range</td>
<td>0.842-0.943</td>
<td>0.855-0.962</td>
<td>0.733-0.952</td>
<td>0.739-0.942</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.945</td>
<td>0.938</td>
<td>0.941</td>
<td>0.925</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.858-0.978</td>
<td>0.815-0.974</td>
<td>0.822-0.977</td>
<td>0.788-0.973</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.993</td>
<td>0.985</td>
<td>0.981</td>
<td>0.979</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.982-0.996</td>
<td>0.979-0.989</td>
<td>0.974-0.984</td>
<td>0.971-0.988</td>
</tr>
</tbody>
</table>

Figure 7. The average SSIM value and overall SSIM numerical range achieved for each combination

<table>
<thead>
<tr>
<th>SSIM</th>
<th>Combination</th>
<th>Gray-Scale Map</th>
<th>Shadow Map</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Label Map</td>
<td>Predicted Image</td>
<td>Ground Truth</td>
</tr>
<tr>
<td></td>
<td>Label Map</td>
<td>Predicted Image</td>
<td>Ground Truth</td>
</tr>
<tr>
<td></td>
<td>Label Map</td>
<td>Predicted Image</td>
<td>Ground Truth</td>
</tr>
</tbody>
</table>

Figure 8. Sample of simulations predicted by Pix2Pix in Gray-scale/ shadow map

5 DISCUSSION AND CONCLUSION

Deep Learning has become an active field of interest by a large portion of designers due to its potentials as well as the advancement in computer resources, efficiency, and algorithms. In our study,
Rhino\Grasshopper has been selected as it is considered one of the most popular tools used by architects and designers, it also provides a wide range of flexibility such as the ability to be integrated with other software (Abdelrahman and Yousef Toutou 2019). In addition, considering the features of wind pressure in façade, we used the shadow map instead of the Gray-scale map aim to increase the efficiency of differentiating the labels. As Figure 7 presented, in both of the Pix2Pix model and the Pix-G model, the shadow map can produce more accurate predictions than the Gray-scale map. Therefore, the shadow map is more effective in representing the depth of each part of the building mass in this research.

In this paper, the proposed method was demonstrated for the early design stages of performance-driven architectural design, employed large amounts of paired images generated by EvoMass and RhinoCFD, and an improved Pix2Pix framework to predict surface wind pressure in high-rise buildings.

Furthermore, we observed that in our experiment the training efficiency of Pix2Pix is dragged down. The advantage of Pix2Pix model is that it uses the GAN framework to provide a general framework for issues of "Image-to-Image translation". In this model, L1+GAN loss is effective at creating realistic renderings that respect the input label maps. However, the model becomes inefficient in the training of the performance predictions, since the Pix2Pix is developed for the issue of image conversion and it needs to sample a large number of feature maps to ensure that the model has strong generalization. Nevertheless, the preference prediction datasets merely contain the label maps of geometrical models and their corresponding simulations, the colors on the simulations represent the same performance data. Therefore, based on this type of image datasets, the L1 loss performs similarly well. Thus, the “Pix-G” model coupled with "shadow map"labeling rule can greatly enhance the accuracy, which provides further insight and new perspective for related research.

6  FUTURE WORK
The presented work of our project enables designers to perform Deep-learning estimations, and predictions. However, the proposed workflow can only generate the predictions in a settled simulation environment. In
order to overcome the limitation, amounts of neural networks could be further trained. Furthermore, more accurate training dataset would be added to the research to reduce the errors of the forecasted results for future research. As the convolutional neural networks have been proven successful in learning geometry representations, the current geometry encoding strategy have better effect in representing the analyses in elevations. However, the feature extraction technology for the performance simulation of the 3D model is currently still in the research stage. This is because geometric figures such as boundaries and geometric parameters can be more easily represented, but those representations are not effective for neural networks since the vectors' semantic meaning varies (Guo et al. 2016). Therefore, future development should also include exploration on geometric representation in 3D models.

ACKNOWLEDGMENTS

This study is funded by National Natural Science Foundation of China (52178017) and China Postdoctoral Science Foundation (2021M701664).

REFERENCES


Guo, X., W. Li, and F. Iorio. 2016. “Convolutional Neural Networks for Steady Flow Approximation”. In 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 481-490.


AUTHOR BIOGRAPHIES

**LIN SUN** is a Ph.D. candidate in School of Architecture and Urban Planning at Nanjing University, China. His research interest includes architectural design and artificial intelligence. His email address is dg1736001@smail.nju.edu.cn.

**SHUQI CAO** is a Ph.D. candidate in School of Architecture and Urban Planning at Nanjing University, China. Her research interest includes computational design and artificial intelligence. Her email address is dg20360001@smail.nju.edu.cn.

**LIKAI WANG** is a postdoctoral researcher in School of Architecture and Urban Planning at Nanjing University. He holds a Ph.D. in Architecture from Nanjing University, China. His research interests include performance-based building design optimization, design exploration, and computational design thinking. He is also the developer of EvoMass, an integrated building massing design generation and optimization tool in Rhino-Grasshopper (https://www.food4rhino.com/app/evomass). His email address is wang.likai@outlook.com.

**GUOHUA JI** is a Full Professor in the School of Architecture and Urban Planning at Nanjing University, China. He holds a Ph.D. in Industrial Engineering from ETH Zurich. He mainly engages in architectural design and methodology, with particular focus on computer-aided architectural design and digital architecture. His email address is jgh@nju.edu.cn.