

DAYLIGHT PREDICTION USING GAN: GENERAL WORKFLOW, TOOL
DEVELOPMENT AND CASE STUDY ON MANHATTAN, NEW YORK

A Thesis

Presented to the Faculty of the Graduate School

of Cornell University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science

by

Mian Jia

May 2021

© 2021 Mian Jia

ALL RIGHTS RESERVED

ABSTRACT

In early design, the building morphology and floor plan determine its daylighting performance. However, climate-based daylight simulations are often computationally expensive and therefore their participation in fast iterative design is limited. Nowadays, image-to-image translation algorithms are promising since climate-based daylighting metrics are often visualized as a floorplan grid that can be represented in a pixel bitmap. The pix2pix, conditional generative adversarial networks (cGANs), can quickly generate corresponding images based on the input images encoded with information. In our work, this method uses pix2pix as a proxy model to provide daylighting results with high accuracy while requiring little computing resources and running in a short time. Hence, using this method, designers can achieve accurate instant daylight performance feedback to polish the design outcome. We have wrapped our work as a new tool in Grasshopper, ArchiGAN, and organized a general workflow to predict daylighting performance for floorplans based on pix2pix.

BIOGRAPHICAL SKETCH

Mian Jia earned the Bachelor of Architecture degree from Southeast University in 2018 and the M.S. degree in Matter Design Computation from Cornell University in 2021. Undergraduate, his main subject is architectural design, but before entering graduate school, he has studied the combination of architecture and cutting-edge technology. He participated in work camps in Vancouver, Rome, Taiwan and Shanghai, and his research included the application of VR in urban design for public participation, cities and big data, and the application of robotic arms in the prefabrication of wooden components. He won an excellent provincial graduation project. And many other awards. In the end, he chose the direction of matter design computation and entered Cornell AAP to study for a master's degree in science. He is passionate about computational design at the intersection of architecture, engineering, and science. He is dedicated to applying his interdisciplinary insights and theories to the practice in urban environment analysis at technological level.

ACKNOWLEDGMENTS

I would like to thank all those who helped me during the writing of this article. First, I would like to express my gratitude to my supervisor Professor Timur Dogan for his constant encouragement and guidance. He led me through all stages of the subject study and the writing of this article. Without his consistent and enlightening guidance, this research and thesis could not have reached its present achievements. Second, I would like to thank Professor Jenny Sabin for his opportunity to participate in the program, which has provided me with abundant resources to support my research. Finally, I would like to thank my beloved family for their love and care for many years and great confidence in me. I also want to express my sincere thanks to my friends and classmates for their help and time to listen to my opinions and help me solve the difficulties encountered in the paper.

TABLE OF CONTENTS

Biographical Sketch	iii
Acknowledgements	iv
Table of Contents	v
1 Introduction	1
1.1 Motivation	1
1.2 Related Work	2
1.3 Our Work	4
2 Methods	6
2.1 Tool development	7
2.2 Voxelization and image encoding	7
2.3 Data sampling	10
2.4 Training	12
3 Results	15
3.1 Pix2pix prediction results	15
3.2 Quantitative analysis	16
4 Conclusion	19
References	20

1 Introduction

1.1 Motivation

Daylighting performance is a very important reference parameter to both urban and architectural design, and it also determines the quality of the design. The intensification of climate change and energy crisis have always been the issues that people are increasingly concerned about. Similarly, in the field of architecture, more and more designers are committed to achieving low-energy green buildings. Natural daylighting is the most energy-saving and environmentally friendly lighting method to improve building energy efficiency [1]. In addition, adequate natural daylight is the key to a comfortable working and living environment, which is essential to the physical and mental health of the residents and users [2].

At the early stage, design and optimization of building morphology greatly affect the overall natural daylighting. Designers mainly consider the overall layout of the design plan, the shape and height of each building, the main openings of the facade and other factors. These are all closely related to the natural daylighting performance. Every design adjustment means a change in the daylighting effect, and conversely, the quality of the daylighting effect can also promote the improvement and optimization of the design. Therefore, instant feedback from daylighting performance will greatly promote the building energy efficiency and indoor comfort of the final design. However, to date, simulation is still time-consuming and computationally expensive. It requires complex inputs and massive computing resources. Also, considering plenty of frequent and repeated changes in the early stage of design, simulation consumes great

amount of extra time, which makes it impossible for designers to frequently use or obtain instant feedback to optimize their designs.

The vigorous development of artificial intelligence and machine learning demonstrates the powerful autonomous learning capabilities of computers in various fields. In the area of computer vision, Generative Adversarial Networks (GAN) is a deep learning model and one of the most promising methods for unsupervised learning on complex distributions in recent years. GAN produces quite good output through mutual adversarial learning among at least two modules in the framework: Generative Model and Discriminative Model [3]. The most common use of GAN is image generation, such as super-resolution tasks, semantic segmentation, and so on. GAN has great potential in daylight prediction because of extremely high efficiency of generating images that show the daylight performance.

Therefore, in this project, we propose a promising, novel method to predict daylight performance based on pix2pix, conditional generative adversarial networks (cGANs). We have developed a Grasshopper tool called ArchiGAN and offered a general workflow for this method.

1.2 Related Work

In the past, many researchers have made plenty of contributions to improve the quality or efficiency of urban daylight simulation. The calculation of the daylit area of the urban designs is used to develop a new tool for designers [4]. This method, as

proven by the researchers, shows simulation results accurately as an almost instant feedback with climate based-daylighting metrics. In addition, simulation-based daylighting analysis is well developed to quantitatively evaluate the daylight performance of the design proposals by computing annual daylighting metrics [5]. These methods are all excellent improvement for implementing simulation-based daylighting analysis to urban design.

Previous improvements are almost based on simulation. The core calculation logic remains unchanged about daylighting performance metrics. However, our work is innovative and evolutionary based on these studies. The core algorithm of our model is image-to-image translation, which means that we use deep learning to let the computer understand how to generate the correct output picture according to the existing conditions.

Pix2pix is a deep learning framework based on conditional GANs, as a general method to solve image-to-image translation problems. These networks not only learn the mapping from the input image to the output image, but also learn a loss function to train the mapping [6]. This makes Pix2pix possible to be implemented in the abundant different tasks about image-to-image translation, such as recoloring the picture or drawing specific objects according to the edges. Specifically, giving an input image with labels in different colors representing walls, windows, or doors, pix2pix model trained from samples can generate a synthesized facade photo. This work inspires a similar workflow showed in Figure 1 to predict expectant visual results from label

maps encoded with information that is related to daylight performance. We also appreciate and acknowledge the daylight application of GANs by T. Galanos [7], whose work proves the feasibility of basic concept. Its general result reveals the great potential of pix2pix in improving efficiency of generating daylighting metric images. It also provides forward-looking advice on how to construct the input label image.

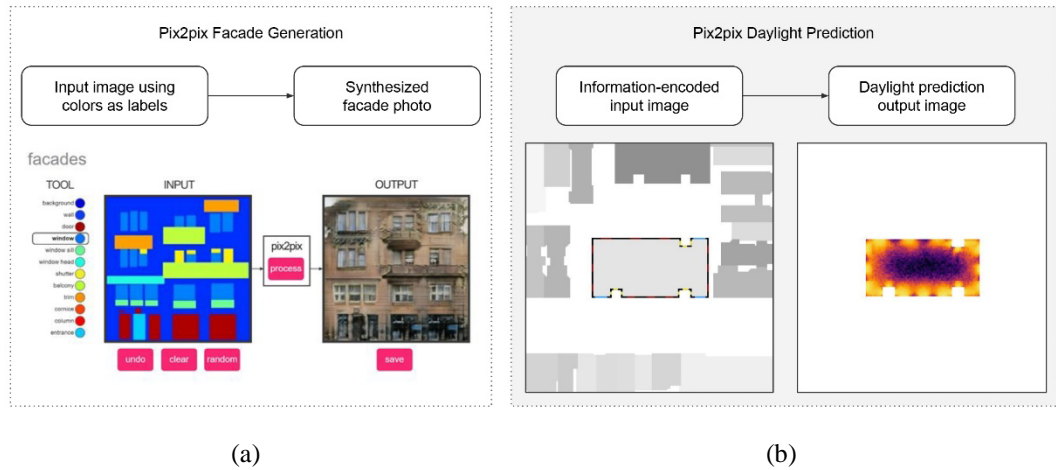


Figure 1 (a) pix2pix example: synthesizing façade photo; (b) inspired daylight prediction method

1.3 Our Work

We believe that this method mainly serves for the early design process, dedicated to optimizing the design by evaluating daylight performance. The purpose of generating images using pix2pix is to greatly shorten the computing time and streamline the workflow, while only relatively high accuracy is expected. Therefore, exploring an applicable general workflow with high efficiency will become the priority in this work.

To apply GANs conveniently and concisely to the architectural field, this work is aimed to not only explore possible general workflows, but also integrate various resources to develop a general tool that can accept GAN models, such as pix2pix. This tool will be directly applied on Rhino Grasshopper platform, allowing designers to use it directly during architecture work and greatly improving the efficiency of the instant feedback. This makes it easier to for designers to take daylight performance into consideration of early design, by achieving the daylight prediction results in an extremely short time.

2 Methods

The general workflow of daylight prediction process based on pix2pix algorithm is illustrated in Figure 2, including dataset preparation, training model and prediction. The first step to prepare the data is convert the 3-dimensional space model into a 2-dimensional label image encoded with information. We use voxelization to convert the geometries into a voxel space, and then we extract the required information and encode using colormaps. For each floorplan, the ground truth of daylight performance is the simulation result calculated by Climate Studio, a fast and accurate environmental performance analysis software used in the field of architecture, engineering, and construction (AEC). The distribution colormap of simulated daylight autonomy will become the expected output image and will be combined with the label input image as a sample. Then, a set of samples generated in the same way will be taken as a dataset to train a pix2pix model and eventually export an ONNX model. To predict, an input label image generated by following the same steps of producing samples can be translated to a colormap of daylight performance by the ONNX model.

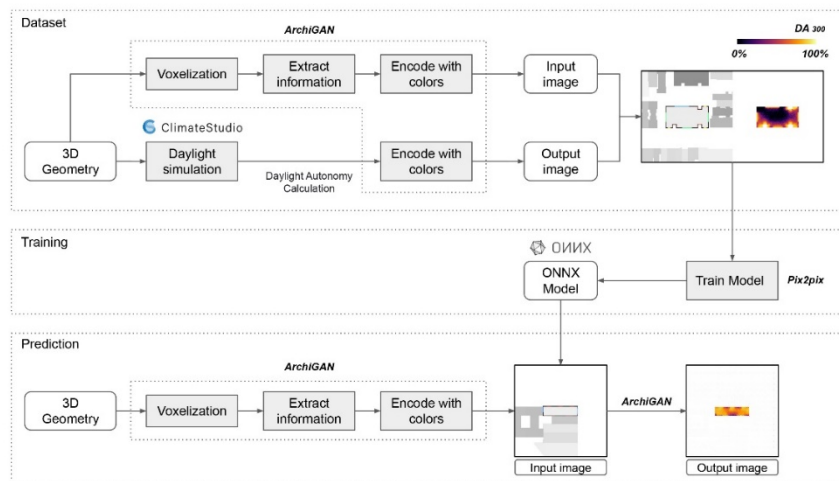


Figure 2 general workflow of daylight prediction based on pix2pix

2.1 Tool development

ArchiGAN, running on Rhino Grasshopper platform, is the core tool we have developed to facilitate the general workflow. It is dedicated to applying image-to-image translation machine learning algorithms to building environment analysis technologies in the field of architecture. It mainly supports functions including transforming 3d geometries into 2d information-encoded images, encoding or decoding information of the images, implementing the Open Neural Network Exchange (ONNX) [8] model for prediction and analyze the prediction accuracy.

As a general toolkit, ArchiGAN can accept multiple kinds of ONNX models that conform to the format of pix2pix generator network. Hence, this tool allows users to use independent models trained on their own datasets to produce expectant visual results in various task scenarios. After training, the model whose effectiveness has been verified can be reused, and it can also be stored and shared for other people to use. In this way, it greatly reduces the waste of computing resources caused by repeated calculations. The sample group for training can be continuously enriched and optimized to strengthen the learning and iteratively produce better models.

2.2 Voxelization and image encoding

In the very beginning, the geometric form is greatly diverse based on the designer's idea. As we have mentioned before, high precision is unnecessary at the early stage of design. To simplify the geometry and extract information conveniently, voxelization is a concise and effective method to transform various types of linear or

non-linear geometry into standardize cubes of the same size in a bounding space [9]. The default cube size is $0.5m*0.5m*0.5m$, and users can modify it to control the precision.

As is showed in Figure 3, points, non-linear lines or meshes can all be converted into standardize cubes. And each cube has a unique coordinate (i, j, k) to provide the location information in the bounding space with the dimension of (X, Y, Z) . Hence, we can easily format all the geometries in the same space into a 2-dimensional image with the resolution of (X, Y) by translating cube coordinates to pixel colors. For each pixel in the image, the coordinates (i, j) store the location of each cube representing building geometries in the space. The coordinates k of all cubes sharing the position (i, j) demonstrates the relative height in the bounding space, which is represented by colors in a certain colormap.

Coordinates of cubes provide different kinds of information from the geometries. As we can see from Figure 4, we can either only record the height of the highest cube or calculate the most height difference for each position (i, j) . Also, we can encode different groups of information with corresponding colormaps. For instance, use color ramp for buildings only and grey scale for the terrain make it easier for the machine to filter and understand the information. In this way, the image can transport various parameters to the machine in the form of pixel colors.

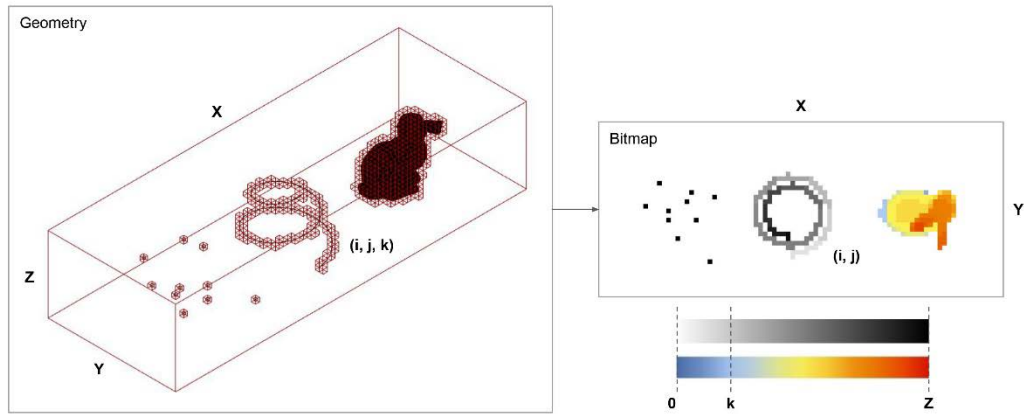


Figure 3 geometry to bitmap: voxelization of geometries and image encoding

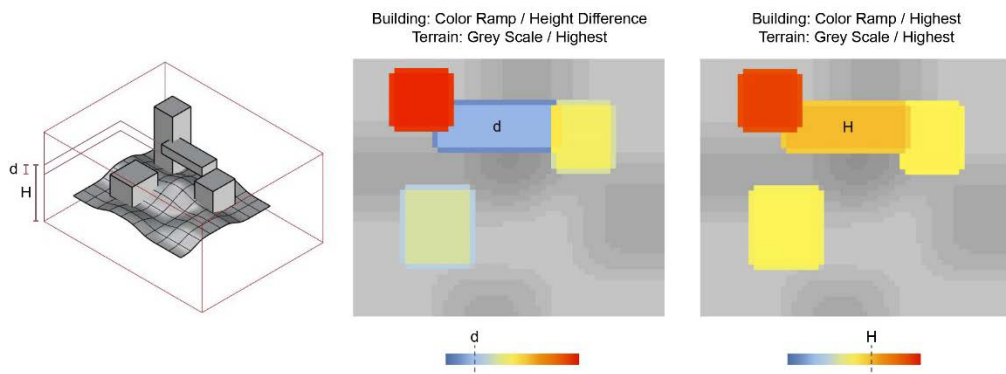


Figure 4 different geometry information encoded using different colormaps

Then, after the voxelization, information from the voxel space of all geometries can be encoded into the image step by step. To train a predictive model for daylighting, the samples should contain the parameters that have huge influence on the indoor daylight performance. Several main impact factors include the local climate, the height difference between the target floor and surrounding buildings, the window-to-wall ratio, and façade materials. To keep the model concise and efficient, façade material will not be taken into consideration since our workflow is for early design.

The local climate condition is also unnecessary to encode since the climate difference of various locations within a certain geographic area is subtle and can be ignored. We can train several models for different places whose climates vary. Figure 5 demonstrates standard steps to produce the input image for daylight prediction. We choose the grey scale colormap to encode the height information of the floor and surrounding buildings. Walls are in black color to clearly define the boundaries between indoor and outdoor. Windows are in colors from Turbo colormap to illustrate the ratio of the windows to the external walls.

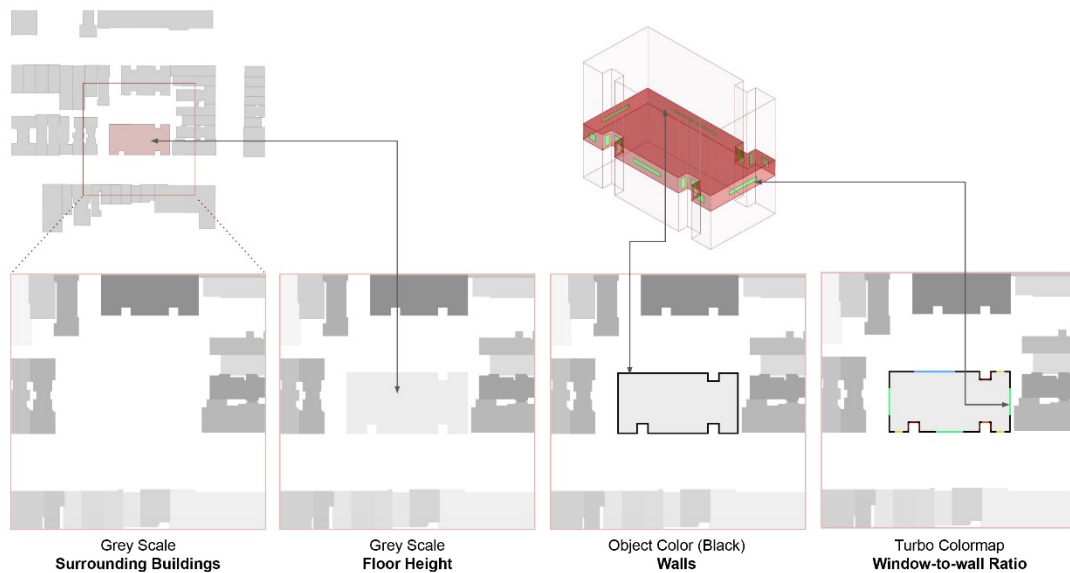


Figure 5 image encoding for a building geometry generated from Manhattan, New York

2.3 Data sampling

To train an effective model, it is of vital importance to prepare a group of valid and sufficient samples. Manually making samples will consume a lot of time and is

not realistic. Searching for samples also suffers great resistance because it is extremely hard to not only find enough matching label images, but also ensure the overall quality. Therefore, in our work, we exploit several components in Urbano [10] and ArchiGAN to develop a general workflow to for data sampling.

This is the general workflow to generate the input label image for a sample for a specific geographical area. Urbano can quickly extract the height information for the target building and all surrounding buildings to form the geometries of an urban environment around the target. For the target building, a component called ‘floor cutter’ in ArchiGAN can divide the geometry to cut out a floor geometry with the expected height. Windows can also be automatically generated according to the size of the walls or options made by users. Therefore, we can have a simple floorplan geometry ready for voxelization to be encoded as a label image.

In this case, we choose Manhattan, New York as a start. Among the island, we have chosen 6 different areas from neighborhoods whose density and morphology are various as shown in Figure 6. Each area provides 50 samples generated diversely and randomly to form a group of 300 samples.

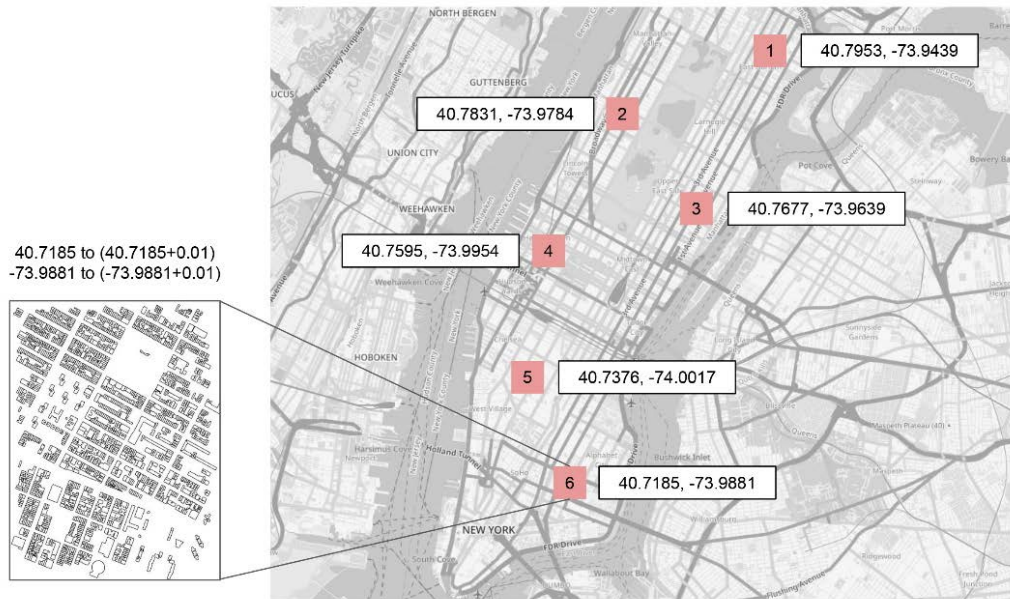


Figure 6 data sampling from Manhattan, New York

2.4 Training

A dataset of samples generated from the general workflow in section 2.3 is divided into a training set of 60%, a validation set of 20% and a test set of 20% to train the Pix2pix model and output the generator (G) and discriminator (D) networks.

The training losses of the G and D converges as displayed in Figure 7 throughout 200 epochs. The curve labeled as D_real and D_fake shows the losses of D for real and fake inputs respectively. According to pix2pix developers, normally the converge of Ds is within the range of 0.1 to 0.3, which is matched in the figure. But it is more important to validate the accuracy of the prediction result to prove the quality of the model.

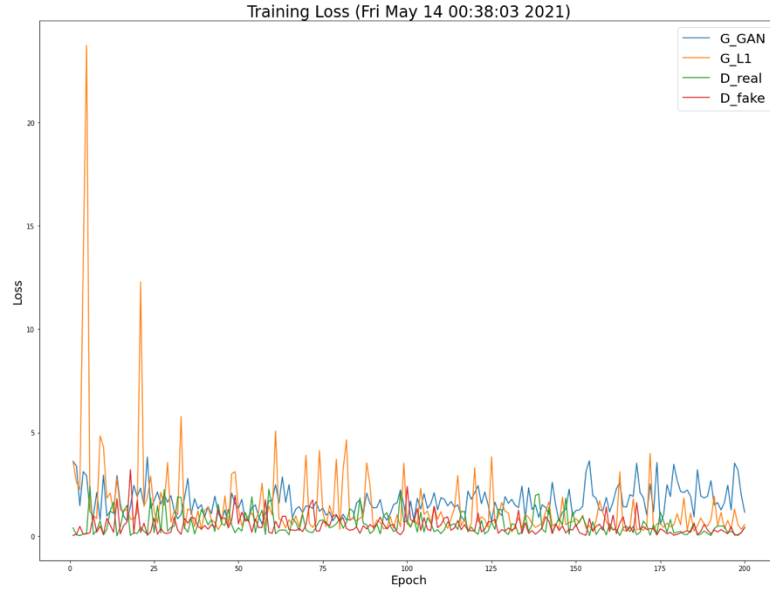


Figure 7 training losses of the G and D throughout 200 epochs

After training, pix2pix outputs all the Gs corresponding to each epoch. Normally the latest one is used as the final G. The network of G is illustrated in Figure 8. Both the input and output of the network are two arrays of $1 \times 3 \times 512 \times 512$ float numbers. They represent the input and output image respectively. 512×512 shows the resolution of the images. For each pixel, the integer numbers of 3 color channels R, G and B are scaled from range $[0, 255]$ to $[-1, 1]$. The input float array will go through the activation functions on layer nodes and become an output float array at the end.

To apply the output generator network in the Rhino Grasshopper, we have modified the training process to save the final network as an ONNX model. The ONNX format is a standard for expressing deep learning models, allowing them to be smoothly transferred between different kinds of frameworks. It supports several

mainstream frameworks including Pytorch, which is the one used in pix2pix model. This facilitates the migration of algorithms and models between different platforms.

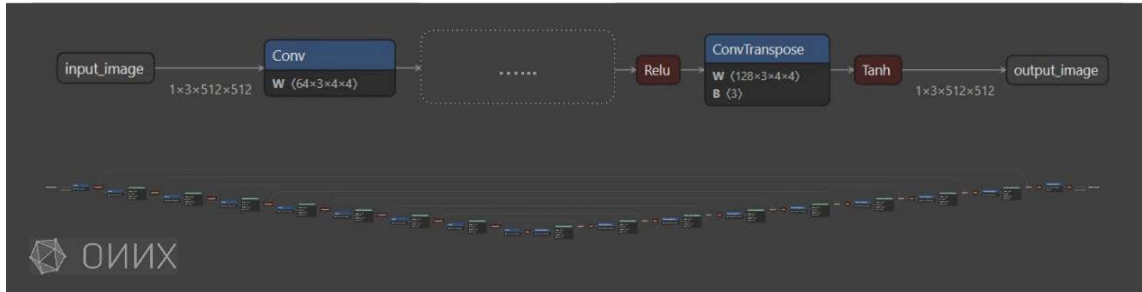


Figure 8 the network of Generator

3 Results

3.1 Pix2pix prediction result

As shown in the Figure 9, the prediction workflow proves to be very time efficient. Once the geometries are ready, voxelization and encoding from the sampling workflow can be applied to produce the corresponding input image. The component ‘predictor’ in ArchiGAN can apply the ONNX model to generate the daylight autonomy distribution map. The whole workflow including input image preparation is almost 2 seconds.

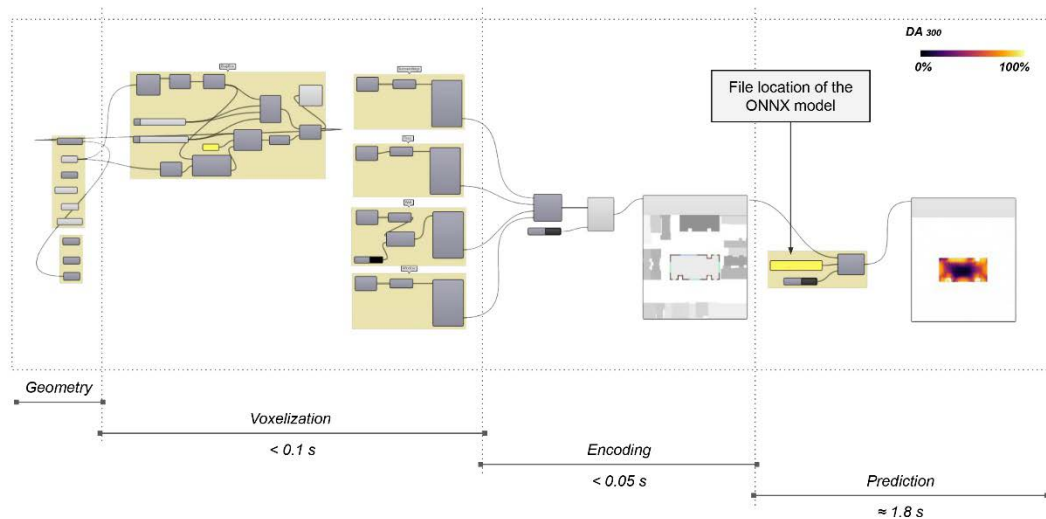


Figure 9 prediction workflow by the pix2pix proxy model

To evaluate the prediction performance of pix2pix model, we use the daylight autonomy image produced from the simulation executed by Climate Studio as the ground truth to compare with the image produced by the predictive proxy model. Figure 10 shows 8 input images from the testing set about the visual quality between simulation and prediction. Basically, we can see that the prediction results are greatly

similar with simulation regarding the naked-eye visual effect. This indicates that pix2pix prediction in early design is feasible.

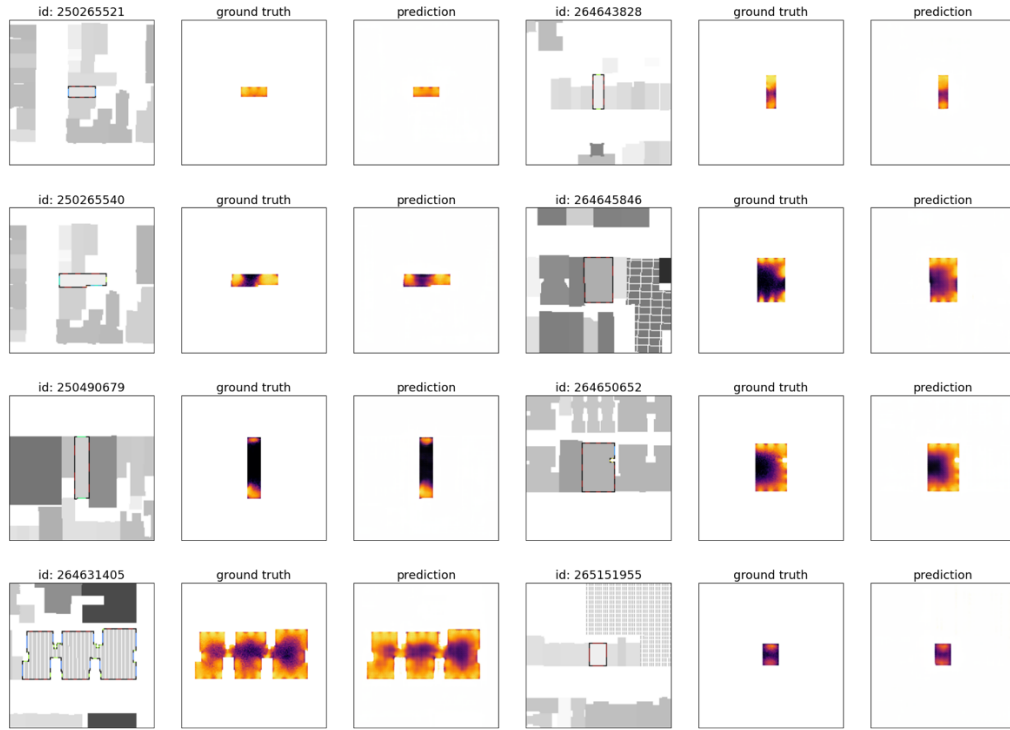


Figure 10 simulation and prediction results of 8 input images from the testing set

3.2 Quantitative analysis

A single comparison with respect to visual effect is not convincing about the validity of the daylighting prediction results. Thus, in our experiment, we record the time cost for both the simulation and prediction workflow. We use image similarity and average DA (300 lux) and to comprehensively evaluate the accuracy of the daylight prediction result. The average value of the ‘color distance’ of the pixels in the ground truth image and prediction image is the similarity of the two pictures. In addition, ArchiGAN supports decoding the colormap information from images to

calculate the average daylight autonomy (DA) value for both simulation and prediction. Daylight autonomy excludes the effects of sunlight and demonstrates an annual estimate of the probability that the daylighting succeeds to achieving or exceeding the required indoor illumination [11].

Figure 11 shows the time cost (seconds) of the daylight simulation and prediction on the testing set. Simulation's time cost is at least 51.0435 seconds and at most 833.6335 seconds, while prediction only cost 1.9190 seconds on average for the whole set. Since the input of pix2pix prediction is a fixed-length float array because of the fixed resolution, pix2pix's time cost is constantly extremely short and very stable. Hence, no matter how complex the geometries are or how big the number of them is, pix2pix model can run prediction with a great time efficiency.

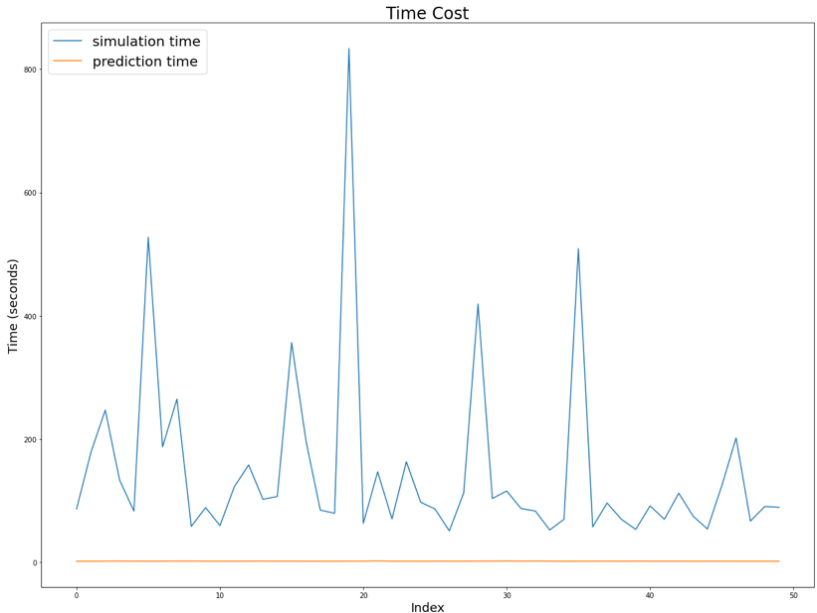


Figure 11 the time cost (seconds) of the daylight simulation and prediction on the testing set

The average of image similarity between simulation and prediction is 91.51%. As shown in Figure 12, the value of similarity remains above 80% among the testing set. On the other hand, from the curves in Figure 13, we can clearly see the DA value provided by simulation and prediction is close to each other. The DA error is only 0.0956 on average throughout the testing set. Evaluation of these two indicators shows great effectiveness and reliable accuracy of pix2pix model in daylight prediction.

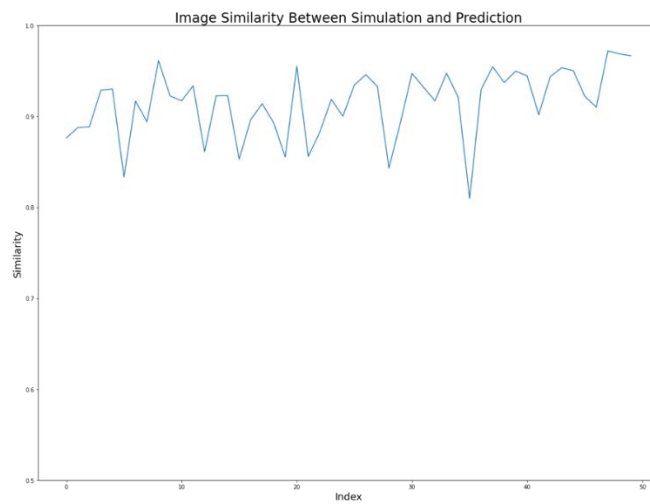


Figure 12 similarity of the simulation and prediction images

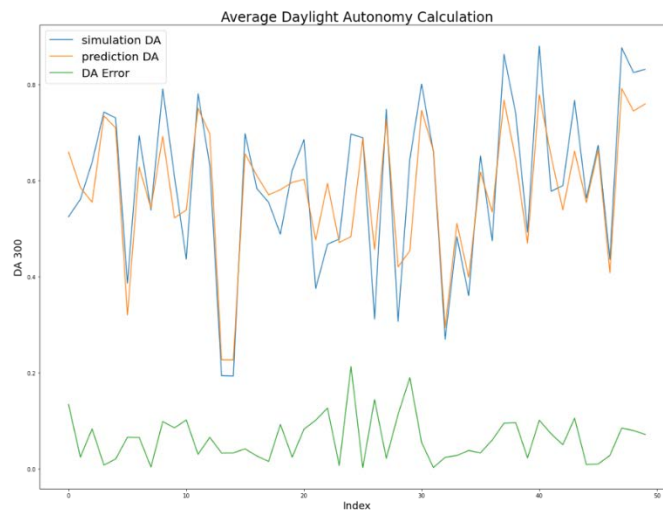


Figure 13 average daylight autonomy of the simulation and prediction results

4 Conclusion

The general workflow of daylight prediction based on the pix2pix GAN algorithm in this work is a novel method that brings deep learning and neural networks to building environment analysis. It provides designers with a fast, simple, effective, and reliable way to obtain the daylight evaluation result of the design plan for optimization at the early stage. This work not only proves the feasibility of leveraging pix2pix in daylight prediction both visually and quantitatively, but also explore and develop a general tool, ArchiGAN, to dock the workflow with the real working scenario. We hope that our work can enable designers to use pix2pix to predict daylight performance without being restricted by time and efficiency, thereby reducing building energy consumption to promote energy conservation and mitigate climate change.

References

- [1] Baker, Nick & Steemers, Koen & Compagnon, Raphaël & Crowther, David & Littlefair, P. & Aschelhoug, O. & Michel, L. & Courret, Gilles & Scartezzini, Jean-Louis & Parpairi, K.. (2021). Daylight Design of Buildings.
- [2] Wirz-Justice, Anna & Skene, Debra & Münch, Mirjam. (2020). The relevance of daylight for humans. *Biochemical Pharmacology*. 114304. 10.1016/j.bcp.2020.114304.
- [3] Goodfellow, Ian & Pouget-Abadie, Jean & Mirza, Mehdi & Xu, Bing & Warde-Farley, David & Ozair, Sherjil & Courville, Aaron & Bengio, Y.. (2014). Generative Adversarial Nets. *ArXiv*.
- [4] Dogan, Timur & Reinhart, C. & Michalatos, Panagiotis. (2012). Urban Daylight simulation Calculating the daylit area of urban designs.
- [5] Saratsis, Emmanouil & Dogan, Timur & Reinhart, C.. (2016). Simulation-based daylighting analysis procedure for developing urban zoning rules. *Building Research & Information*. 45. 10.1080/09613218.2016.1159850.
- [6] Isola, Phillip & Zhu, Jun-Yan & Zhou, Tinghui & Efros, Alexei. (2017). Image-to-Image Translation with Conditional Adversarial Networks. 5967-5976. 10.1109/CVPR.2017.632.
- [7] Personal project by Theodore, Galanos using GANs to streamline daylighting analysis in the AEC. <https://www.mitevpi.com/daylight-gan>
- [8] ONNX. Open Neural Network Exchange, an open format built to represent machine learning models. <https://onnx.ai/>, 2017 (accessed April 5, 2020).
- [9] Chen, Min & Kaufman, Arie & Yagel, Roni. (2000). Volume Graphics. 10.1007/978-1-4471-0737-8.
- [10] Dogan, Timur & Yang, Yang & Samaranayake, Samitha & Saraf, Nikhil. (2020). Urbano: A Tool to Promote Active Mobility Modeling and Amenity Analysis in Urban Design. *Technology | Architecture + Design*. 4. 92-105. 10.1080/24751448.2020.1705716.
- [11] Paule, B. & Pantet, S. & Boutillier, J. & Sergent, CH & Valentin, E. & Roy, Nicolas. (2013). Diffuse Daylight Autonomy Towards New Targets.