Performance Prediction of AI-generated Architectural Layout Design: Using Daylight Performance of Residential Floorplans as an Example

Xiao Hu[1], Hao Zheng[2], Dayi Lai[3]

Abstract: The integration of artificial intelligence (AI) in architectural design, especially for generating floor plans, can greatly streamline the process. However, most AI-generated plans focus on form and spatial layout, often neglecting crucial performance evaluations because they are presented as images without the necessary geometric and physical properties for effective simulation. To address this limitation, we propose a novel approach that combines diffusion models with generative adversarial networks (GANs) for generating and evaluating floor plans. We fine-tuned a Low-Rank Adaptation (LoRA) model for creating residential floor plans, while a GAN quickly predicts daylighting performance. Our results show that the diffusion model generates a more varied set of floor plans compared to the training set. The GAN accurately assesses daylighting performance, with deviations from the ground truth not exceeding 5%, achieving a mean squared error (MSE) of 4.2 and a structural similarity index (SSIM) of 0.98. Additionally, it operates 267 times faster than traditional methods. This approach equips architects with a reliable tool for efficient early-stage design decisions, enhancing AI-driven workflows.

Keywords: Automated floor plan, Diffusion model, Generative design, Conditional generative adversarial network

1 Introduction

1.1 Research Background

The use of automated tools in architectural floor plan generation, especially for residential design, is gaining traction due to their ability to enhance efficiency by reducing repetitive tasks and minimizing trial and error in the design process (20). Automated floor plan generation typically utilizes two main methods: rule-based and learning-based approaches (15). Rule-based methods rely on algorithmic constraints derived from design principles, using predefined generative rules, shape grammar (18), and advanced techniques like agent-based modeling (1,5) and graph theory (2,12). However, these often require architects to possess specialized skills in mathematics and programming, creating barriers to widespread adoption. In contrast, learning-based methods have gained traction due to their accessibility and ease of implementation, allowing architects to bypass complex algorithms by extracting features from existing data. The rise of machine learning and AI has led to data-driven approaches that learn from extensive design datasets, resulting in diverse and creative floor plans. Techniques like Generative Adversarial Networks (GANs) (4), supervised learning with Auto-Encoder refinement (21), and Graph Neural Networks (GNNs) (14) exemplify this trend, making it increasingly common for architects to integrate AI tools into their workflows for efficient floor plan generation.

Despite advancements in automating floor plan generation, the evaluation of the generated designs is still lacking, particularly in the context of learning-based approaches. While rule-based methods facilitate the straightforward definition and adjustment of design variables for performance optimization(22), learning-based

[1] Xiao Hu, Shanghai Jiao Tong University, Minhang, 200240 Shanghai, China, e-mail: huxiao1009@sjtu.edu.cn

[2] Hao Zheng (🖂), City University of Hong Kong, Kowloon Tong, Hongkong, China, e-mail: hazheng@cityu.edu.hk

[3] Dayi Lai (🖂), Shanghai Jiao Tong University, Minhang, 200240 Shanghai, China, e-mail: dayi_lai@sjtu.edu.cn

methods often embed these variables within generated images, thereby complicating traditional evaluation techniques. This necessitates the use of specialized software for geometric modeling and simulation, which can be time-consuming and requires expertise. To address this gap, recent developments in deep learning, particularly with Generative Adversarial Networks (GANs) like pix2pix (10), show promise in converting generated images into actionable performance data.

Nevertheless, the incorporation of performance assessment into the domain of learning-based architectural design has been largely neglected. This paper puts forth innovative model architectures and strategies that integrate generation with performance evaluation, leveraging diffusion models and conditional GANs (cGANs) to enhance the usability of AI-generated floor plans and equip architects with invaluable tools for early-stage design decisions.

1.2 Literature Review

Diffusion models are emerging as a powerful alternative to GANs in automated floor plan generation. While GANs, have been widely used for image generation, diffusion models like Denoising Diffusion Probabilistic Models (DDPMs) (8) and Latent Diffusion Models (LDMs) (16) offer enhanced stability and diversity. These models can leverage textual descriptions for guided image generation, expanding their applicability. Recent studies have demonstrated the potential of diffusion models in architectural design, with innovations like HouseDiffusion (17) effectively generating vector floor plans that maintain geometric relationships, and FloorplanDiffusion (23) allowing multi-conditional inputs to produce designs. However, a key limitation of these models is their neglect of environmental performance factors such as daylighting and ventilation, which are critical for practical applications in early-stage design.

In the realm of performance prediction, deep learning methods, particularly Convolutional Neural Networks (CNNs) and GANs, have streamlined the assessment of building performance, which is essential for creating efficient and sustainable environments. Traditional assessment methods can be complex and time-consuming, but DL offers efficient alternatives. For instance, multimodal GANs have reduced computational time for daylight prediction significantly (11), while CNNs have provided real-time feedback on daylight performance in floor plans (6). Additionally, GANs have shown promise in Computational Fluid Dynamics (CFD) for urban airflow prediction and optimizing environmental factors(13). Despite these advancements, integrating performance evaluation with AI-generated designs remains largely unexplored, presenting an opportunity for further research to enhance the efficiency of both design and assessment processes.

1.3 Goal of This Study

The goal of this study is to create a streamlined workflow that integrates performance optimization into AI-generated residential floor plans. This research addresses the complex relationships between design generation and environmental performance by combining the generative strengths of diffusion models with the performance evaluation capabilities of GANs. The main contribution is an automated system that generates diverse and functional floor plans while providing real-time feedback on daylight performance, facilitating rapid design iterations. By merging design generation with performance evaluation, this workflow enhances the practicality and effectiveness of AI-driven design in architectural practice, allowing for more informed decision-making.

2 Methodology

2.1 Overall framework

The overall framework enables designers to quickly generate and evaluate architectural floor plans using a workflow that combines a diffusion model and a conditional GAN (cGAN), specifically the pix2pixHD (19) model. Architects start by creating floor plan layouts from text prompts with the diffusion model. These layouts are then analyzed by the cGAN to predict their performance. The process involves two main steps: first, a rule-based parametric model is developed using Rhino and Grasshopper to create floor plans that adhere to specific spatial requirements, generating a dataset for the diffusion model. Second, the Ladybug and Honeybee plug-ins perform simulations to produce daylight performance data, which, along with the floor plan images, forms the training dataset for the pix2pixHD model.

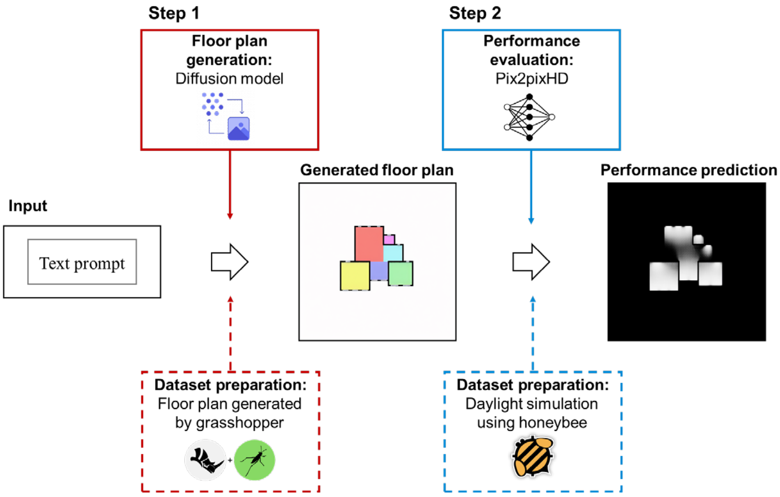


Fig. 1 Overview of the workflow.

2.2 Floor Plan Generation

In this study, we use the diffusion model to generate building floor plans. We fine-tune the diffusion model using the Low-Rank Adaptation (LoRA) model(9), which is designed to adapt large pretrained models to new datasets without requiring full retraining from scratch. To prepare the dataset, we use a parametric algorithm to generate floor plans.

2.2.1 Dataset for Training LoRA

In the process of generating floor plans, a dataset is required for training. A parametric algorithm was developed to generate floor plans as a dataset. The method combines different rooms by applying forces between them and using the final balance of forces to create a tightly connected floor plan. As illustrated in Figure 2, the creation of a residential floor plan entails considering the number of rooms, the area of each room, and their topological relationships. The process comprises two main phases: room generation and force application to nodes.

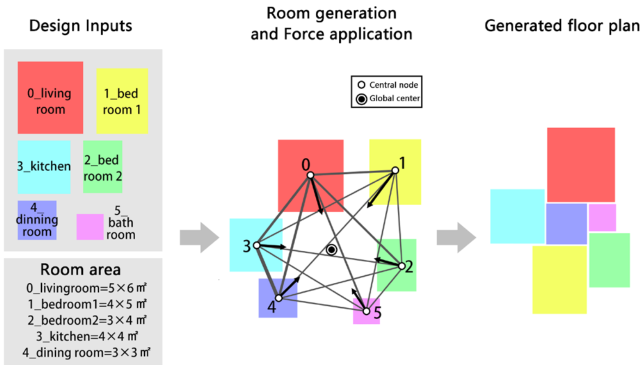
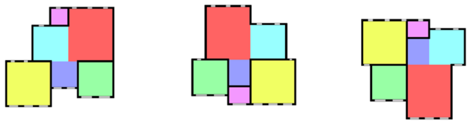
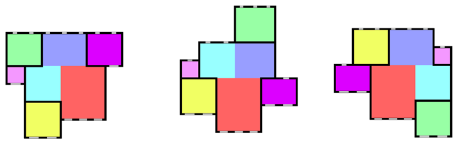


Fig. 2 Generating floorplans through parametric generation algorithm

Room generation phase: the coordinates (x_i, y_i) of the central node determine each room's initial position. Rectangular shapes are created with the central node as the center, representing each room's designated area.

Force application phase: forces are applied to nodes in two steps. First, a global center point (typically the origin) is established, and an attractive force clusters all room nodes around this center. Then, varying spring forces between room nodes ensure their topological relationships match the design inputs. For instance, dining rooms and kitchens typically have strong connections, so the spring forces between their nodes are set very high. If these rooms are not adjacent, the spring forces will attempt to move them closer together. The dataset generated by the script and used to train LoRA is shown in Table. 1.

Table. 1 Dataset used to train LoRA

Type	Data volume	Area/m^2	Cases from dataset	Text tag
5_rooms	320	50		
6_rooms	320	100		floor-plan_XH;
7_rooms	320	120		
Legend of room types			**Living room** (255,99,99); **Kitchen** (152,255,255); **Dining room** (152,158,255); **Other room** (220,0,255); **Bed room1** (241,255,99); **Bed room2** (152,255,165); **Bath room** (233,152,255)	

2.2.2 Diffusion Model and LoRA Training

Diffusion models work by adding Gaussian noise to training data (forward diffusion process) and then learning to reverse this process step by step to recover the original data (reverse diffusion process). Diffusion-based image generators have seen widespread commercial interest, such as Stable Diffusion and DALL-E. These models typically combine diffusion models with other models, such as text-encoders and cross-attention modules to allow text-conditioned generation. In this study, we use stable diffusion to finish our tasks.

Low-Rank Adaptation (LoRA) is trained to adapt the weights of Stable Diffusion to learn floor plan layout patterns. In deep learning models, weight matrices W are often high-dimensional. For instance, in fully connected layers or convolutional layers, the dimensions of the weight matrices can be very large. Given a weight matrix $W \in \mathbb{R}^{d\times k}$, LoRA approximates the weight update ΔW as a product of two low-rank matrices:

$$\Delta W = AB^T \qquad (1)$$

where $A \in \mathbb{R}^{d\times r}$ and $B \in \mathbb{R}^{k\times r}$, and r is much smaller than both d and k. The updated weight matrix W' is then given by:

$$W' = W + \Delta W = W + AB^T \qquad (2)$$

The primary objective of LoRA is to minimize the loss function L on the training data, where the model parameters are updated using the low-rank adaptation:

$$\min_{A,B} L = MSE\left(W_{pretrained} + AB^T\right) \qquad (3)$$

Here, $\theta_{pretrained}$ represents the pre-trained model parameters, and A and B are the low-rank matrices. Gradient descent is used to update the parameters of the low-rank matrices A and B.

2.3 Performance Evaluation

To evaluate the daylight performance of floor plan images generated by diffusion models, the traditional method involves importing images into professional modeling software, establishing geometrical and physical models, applying materials, and running simulations. Current image-to-image prediction methods offer a solution to this problem. Therefore, this study employs the pix2pixHD model, known for its strong performance in image-to-image translation, to efficiently evaluate the results generated by the diffusion model.

2.3.1 Dataset for Training Pix2pixHD

Daylight simulation was conducted using Ladybug and Honeybee on the Grasshopper platform in Rhino to generate image pairs for training the pix2pixHD model. We used the EPW weather file for Shanghai. The building geometry features window-to-wall ratios of 0.2 for the north-facing façade and 0.3 for the south-facing façade. The simulation parameters conform to the *Code for Thermal Design of Civil Buildings* (GB50176-2016)(3) and adhere to standard construction dimensions. Material properties were selected to reflect commonly used construction materials, as detailed in Tables 2 and 3.

Table. 2 Simulation parameters

Parameters	value
Window to wall ratio(north)	0.2
Window to wall ratio(south)	0.3
Windows height	1.8m
sill level height	0.9m
Story height	3.3m

Table. 3 material property in daylight simulation model

Opaque Material	Reflectance
Ceiling	0.6
Wall	0.5
Floor	0.3
Glazing material	**Transmissivity**
Windows glaze	0.6

Spatial Daylight Autonomy (sDA) measures the percentage of a space that receives adequate daylight throughout the year. According to the Illuminating Engineering Society (IES), sDA300/50% indicates the proportion of points in an area that achieve a horizontal illuminance of at least 300 lx for 50% of occupied hours (8 a.m. to 6 p.m.) over a typical meteorological year. The sDA equation is as follows:

$$sDA_{x/y\%} = \frac{\sum_i^n P(i)}{\sum_i^n p_i} \in [0,1]$$
$$where\ P(i) = \begin{cases} 1\ ;\ if\ DA \geq DA_{limit} \\ 0\ ;\ if\ DA \leq DA_{limit} \end{cases} \tag{4}$$

The sDA standard, as defined by IES Lighting Measurements (LM) 83-12 (7), uses the parameters "sDA 300,50%" measured from 8 AM to 6 PM over the year. This metric evaluates the daylight performance of a space by categorizing it into three classes based on the percentage of the area meeting the sDA criterion, the details are listed in Table. 4:

Table. 4 standard of spatial daylight autonomy

Level of recommendation	Minimum target illuminance/lx	Fraction of daylight hours/%	Fraction of spaces for target level/%
preferred			≥75%
nominally accepted	300	50%	55%~75%
low			≤55%

2.3.2 Model Training of Pix2pixHD

With the training dataset, we trained a pix2pixHD model to quick predict the daylight performance, which consists of two components: a coarse-to-fine generator and multi-scale discriminators. The generator of pix2pixHD is called coarse-to-fine generator, which can be decomposed into sub-networks, a global generator network G_1 and a local enhancer G_2. The global generator network G_1 is responsible for generating a coarse, low-resolution version of images. This network captures the overall structure and global features of the image but does not focus on fine details. The output from G_1 serves as the foundational layer upon which further refinements are made. To evaluate the generated images, pix2pixHD employs multi-scale discriminators at different scales. Each discriminator is responsible for assessing the realism of the image at a specific resolution, ensuring that both local details and the global structure are realistic.

The authors of Pix2PixHD improved upon the traditional conditional GAN loss $\mathcal{L}_{GAN}$ by incorporating the Feature matching loss $\mathcal{L}_{FM}$ and perceptual loss $\mathcal{L}_{VGG}$. The complete loss function consists of the following equations:

$$\mathcal{L}_{total}(G, D_k) = \sum_k \mathcal{L}_{GAN}(G, D_k) + \lambda_{FM} \sum_k \mathcal{L}_{FM}(G, D_k) + \lambda_{VGG}\mathcal{L}_{VGG}(y, G) \qquad (5)$$

In this study, a total of 300 pairs of images were used for training, of which 80% were used for the training set and 20% for the test set. The pix2pixHD model was trained using the Adam optimizer with a learning rate of 0.0002, due to the time required per epoch, which spans several minutes, the maximum number of epochs was set to 200 to balance performance and computational cost. Both networks were updated at every step using loss functions, which guided the model to generate realistic data through iterative optimization. The loss weights were set to λ_{FM} = λ_{VGG} = 10.

2.3.3 Model Evaluation

To quantitatively determine the optimal epoch, we calculated the Fréchet Inception Distance (FID) scores for each epoch. The Fréchet Inception Distance (FID) is a metric used to assess the quality of generated images by comparing their statistical properties to those of real images. FID calculates the distance between the feature distributions of the generated images and real images, as extracted by an Inception v3 network. Lower FID scores indicate that the generated images are more similar to the real images.

After selecting the optimal model, we evaluate its performance using three metrics: Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM). These metrics provide a comprehensive evaluation by measuring the pixel-level accuracy, perceptual quality, and structural fidelity of the generated images.

3 Results

3.1 Training Results of LoRA

Fig. 3 illustrates the loss curve during LoRA training, spanning 48,000 steps across 10 epochs. A consistent downward trend in the loss indicates effective learning and adaptation by the model. The curve shows a steep decline in the initial epochs, gradually tapering off in later epochs, suggesting convergence. Minor fluctuations in the loss values, due to inherent variability in the training data and the stochastic nature of the AdamW8bit optimizer, remained within acceptable limits, demonstrating the stability and robustness of the training process. Notably, the loss reached its lowest point at approximately the 40,000th step (step 39,731). Given that each epoch consists of 4,800 steps, we selected the 8-th epoch for generating the building floor plan.

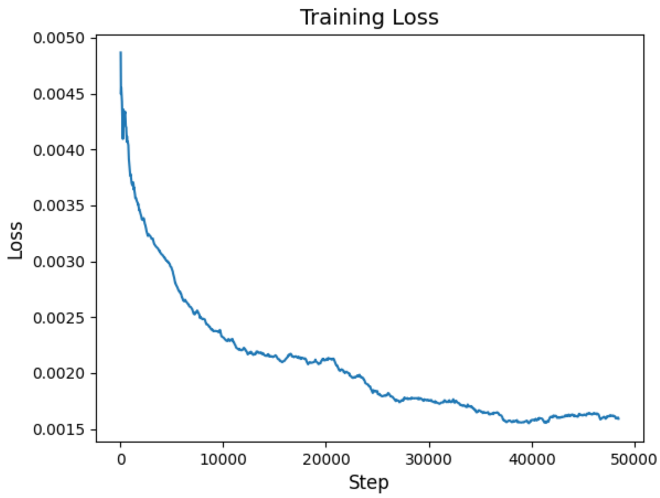


Fig. 3 Traning loss of LoRA

3.2 Training Results of Pix2pixHD

Fig. 4 displays pairs of images at various training epochs for the model. We compared the synthesized images from epochs 10, 30, 50, 100, 150, and 200 with their corresponding ground truth images. Starting from epoch 30, the differences between the synthesized and ground truth images become difficult to distinguish through visual inspection.

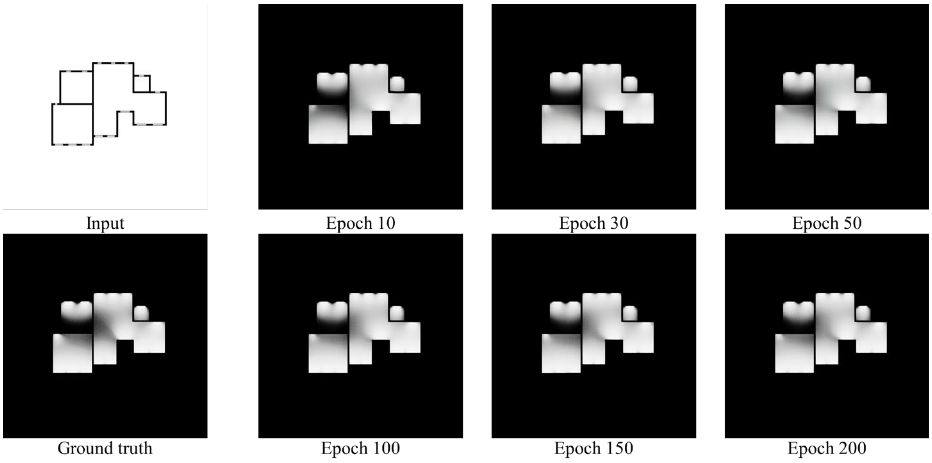


Fig. 4 Daylight performance prediction using pix2pixHD under different epochs

shown in Fig. 5, we found that the model reached its lowest FID score of 12.85 at epoch 170. This indicates the best performance in terms of image quality and similarity to the ground truth. Consequently, the model from epoch 170(170_net_G) was selected for subsequent predictions, ensuring the highest fidelity in generated results.

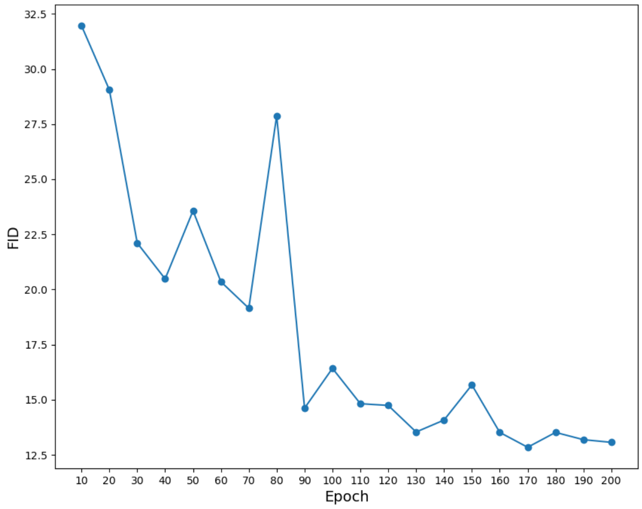


Fig. 5 FID score of pix2pixHD model at different epochs

Table. 5 presents the performance evaluation of the selected model on both the training set and the test set. The model shows consistently strong results, with an average SSIM of 0.98 for both the training and test sets, approaching the ideal value of 1, which indicates a high level of structural similarity with the ground truth. The MSE scores are low, averaging 4.9 on the training set and 4.2 on the test set, reflecting minimal prediction error. Correspondingly, the PSNR scores, derived from MSE, are around 40 dB on the test set, signifying high-quality image generation.

Table. 5 Performance Metrics (MSE, PSNR, SSIM) on Training and Test Sets

Dataset	Average of MSE	Average of PSNR	Average of SSIM
Training set	4.9	41.8	0.98
Test set	4.2	42.6	0.98

As shown in Fig. 6, to verify the accuracy of the model, we compare the predicted values generated by pix2pixHD with the sDA simulation results, and find that the predicted values deviate from the actual values within 5%, with a standard deviation of 2.08, which proves that our trained model is able to predict the daylight performance of planar maps well.

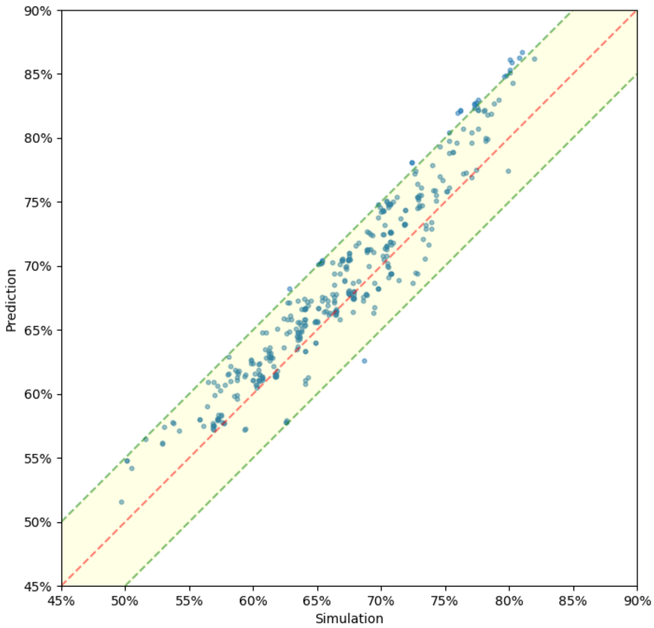


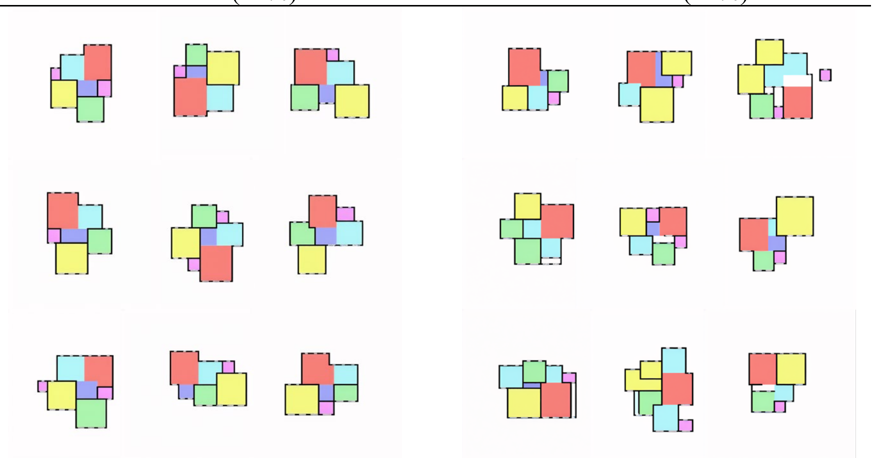
Fig. 6 Comparison of sDA of predicted values with ground truth

3.3 Floor Plans Generation and Evaluation

To evaluate the generation model's effectiveness, we produced 4,800 layouts, which were filtered using an OpenCV script. As shown in Table. 5, these layouts, each at a resolution of 1024x1024 pixels, were created by the diffusion model. After filtering, 89% closely matched the training set features and met key design constraints, while the remaining layouts had issues like misplaced windows, duplicated spaces, or incomplete partitions. Although post-processing could resolve these errors, this study primarily focuses on the diffusion model’s ability to generate plausible layouts for early-stage design. The high success rate underscores the model's capability to capture essential architectural features, making it a valuable tool for rapid preliminary design in building development.

Table. 1 samples of generated results

Good case (89%)	Bad cases (11%)



The floor plans generated by the diffusion model were assessed for Spatial Daylight Autonomy (sDA) using the pix2pixHD model. Most plans fell within the nominally accepted range (55% < sDA $\leqslant$ 75%), with 2,787 cases (65%) meeting this criterion. Conversely, 1,087 plans (25.3%) showed low daylight performance (sDA $\leqslant$ 55%), while only 415 plans (9.7%) achieved preferred performance (sDA > 75%). This distribution highlights the significant impact of floor plan layout on daylight performance, emphasizing the potential of automated generation tools to optimize designs. Fig. 7 shows that as floor area increases, the percentage of plans achieving preferred sDA performance decreases, largely due to the complexities of larger layouts. For architects, automated tools can efficiently generate diverse design options, facilitating refinements based on daylighting outcomes.

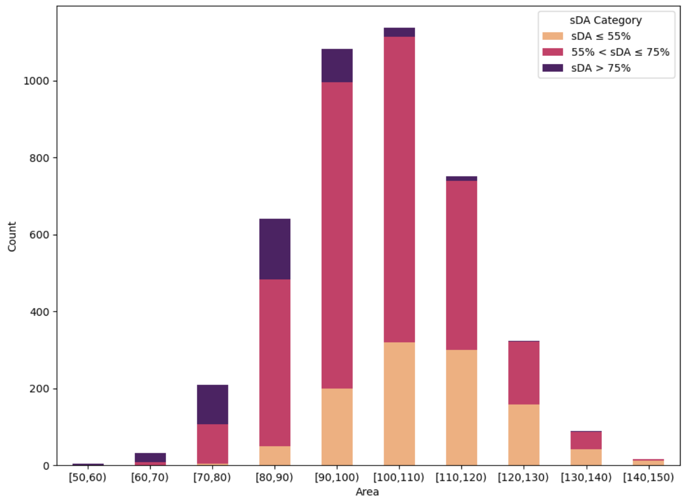
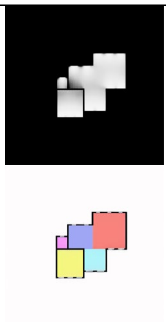
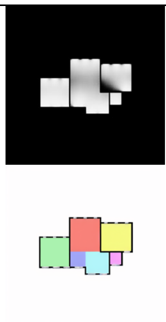
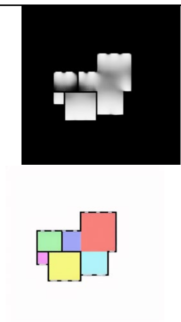
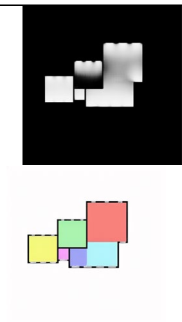
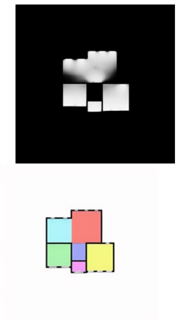
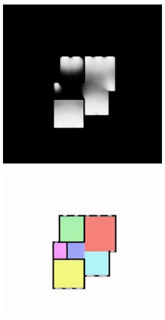
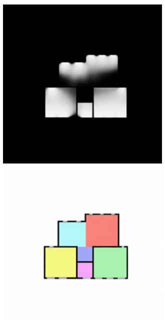
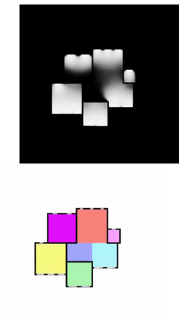
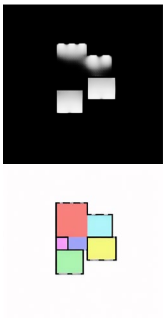
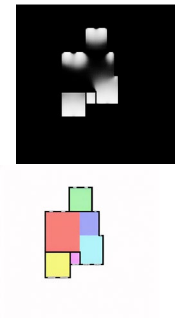
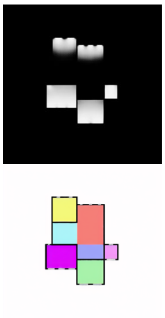
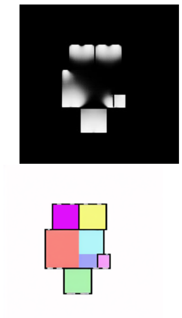


Fig. 7 sDA distribution of generated result

To evaluate the impact of spatial layout on daylight performance, representative floor plans were selected from four size ranges: [80,90), [90,100), [100,110), and [110,120). "Preferred" plans maximize daylight with a horizontal layout featuring north-south windows, while "nominally accepted" plans are more compact, sometimes resulting in inadequate lighting. "Low-performance" plans, with vertically stacked rooms, suffer from poor daylight due to limited east-west windows. Although higher spatial Daylight Autonomy (sDA) is desirable, achieving over 75% sDA isn't always necessary. Wider, shallower layouts optimize natural light but can increase wind loads and reduce insulation, whereas narrower designs may limit lighting and ventilation. Most designs strike a moderate balance, though room layout variations can significantly affect performance. This basic analysis highlights the need for nuanced approaches that consider specific room requirements, suggesting future research should evaluate daylight performance based on occupants' daily routines.

Table. 2 Representative results of AI generation

area	80-90 m^2	90-100 m^2	100-110 m^2	110-120 m^2
Preferred (sDA≥75%)	 sDA=83%	 sDA=82%	 sDA=80%	 sDA=76%
Nominally accepted (75%≥sDA≥55%)	 sDA=66%	 sDA=63%	 sDA=60%	 sDA=58%
Low (sDA≤55%)	 sDA=51%	 sDA=44%	 sDA=44%	 sDA=40%

4 Discussion

Architectural design is complex, requiring a balance of multiple stakeholders' needs, and cannot be fully replaced by automated layout generation tools. However, these tools excel in providing feedback, design guidance, and optimization during the initial stages by generating various design options and identifying promising solutions. Our method generates relatively simple floor plans that offer essential layout information, facilitating relevant guidance in the early-stage design. With just a basic text prompt, it simplifies the design process and prioritizes early-stage guidance focused on performance and environmental factors. Notably, our approach demonstrates significant efficiency, as the pix2pixHD model evaluates performance 267 times faster than traditional daylight simulation methods.

Table. 6 Time costs of floorplan generation and performance evaluation

Task	Test	Training time	Calculation time per case
Generation of floor plans	Rhino + Grasshopper	N/A	15~30 s
	Diffusion model + LoRA (1024*1024)	353 mins	12 s
Performance evaluation	Daylight simulation (grid size: 0.1)	N/A	6 mins 14s (374s)
	Pix2pixHD	423 mins	1.4 s

Despite its advantages, our method has limitations. Improved controllability and reliability in floor plan generation are needed, particularly regarding room types, sizes, and finer architectural details like window placement. Additionally, the current performance prediction is limited to specific and simple floor plan styles, affecting the generalizability of the methodology.

5 Conclusion and Future Work

This study integrates diffusion models and GANs to optimize AI-generated architectural floor plans, shifting from a form-driven to a performance-driven approach that better meets architects' practical needs. By significantly reducing the time for iterative design modifications and performance simulations, the method improves efficiency in the early stages of design.

We implemented performance evaluation using the pix2pixHD model, allowing rapid predictions of daylight performance. This integration combines AI's creative generation with performance evaluation, providing architects with informed design solutions that meet both creative and performance criteria. Our learning-based approach offers a remarkable speed advantage over traditional rule-based methods, reducing performance simulation time from 6 minutes and 14 seconds to just 1.4 seconds—an impressive 267-fold improvement.

However, our model faces limitations regarding the controllability of generative outputs and the generalizability of predictions. Future work will focus on incorporating real-time interaction tools and expanding the variety of floor plan types to enhance effectiveness and applicability.

Reference

1. Arvin SA, House DH. Making Designs Come Alive: Using Physically Based Modeling Techniques in Space Layout Planning. In: Augenbroe G, Eastman C, editors. Computers in Building [Internet]. Boston, MA: Springer US; 1999 [cited 2024 Jun 22]. p. 245–62. Available from: http://link.springer.com/10.1007/978-1-4615-5047-1_16
2. Bao F, Yan DM, Mitra NJ, Wonka P. Generating and exploring good building layouts. ACM Trans Graph. 2013 Jul 21;32(4):1–10.
3. GB50176 C. Code for thermal design of civil building. Ministry of Housing and Urban-Rural Development of the People's Republic of China: Beijing, China. 2016;
4. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Commun ACM. 2020 Oct 22;63(11):139–44.
5. Guo Z, Li B. Evolutionary approach for spatial architecture layout design enhanced by an agent-based topology finding system. Frontiers of Architectural Research. 2017 Mar;6(1):53–62.
6. He Q, Li Z, Gao W, Chen H, Wu X, Cheng X, et al. Predictive models for daylight performance of general floorplans based on CNN and GAN: A proof-of-concept study. Building and Environment. 2021 Dec;206:108346.
7. Heschong L, Wymelenberg VD, Keven, Andersen M, Digert N, Fernandes L, et al. Approved Method: IES Spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE). In 2012. Available from: https://api.semanticscholar.org/CorpusID:114427665
8. Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. Advances in neural information processing systems. 2020;33:6840–51.
9. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, et al. LoRA: Low-Rank Adaptation of Large Language Models [Internet]. arXiv; 2021 [cited 2024 Aug 24]. Available from: http://arxiv.org/abs/2106.09685
10. Isola P, Zhu JY, Zhou T, Efros AA. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017. p. 1125–34.
11. Li X, Yuan Y, Liu G, Han Z, Stouffs R. A predictive model for daylight performance based on multimodal generative adversarial networks at the early design stage. Energy and Buildings. 2024 Feb;305:113876.
12. Merrell P, Schkufza E, Koltun V. Computer-generated residential building layouts. ACM Trans Graph. 2010 Dec;29(6):1–12.
13. Mokhtar S, Sojka A, Davila CC. Conditional generative adversarial networks for pedestrian wind flow approximation. In: Proceedings of the 11th annual symposium on simulation for architecture and urban design. 2020. p. 1–8.
14. Nauata N, Hosseini S, Chang KH, Chu H, Cheng CY, Furukawa Y. House-GAN++: Generative Adversarial Layout Refinement Network towards Intelligent Computational Agent for Professional Architects. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) [Internet]. Nashville, TN, USA: IEEE; 2021 [cited 2024 Sep 13]. p. 13627–36. Available from: https://ieeexplore.ieee.org/document/9577959/
15. Rahbar M, Mahdavinejad M, Markazi AHD, Bemanian M. Architectural layout design through deep learning and agent-based modeling: A hybrid approach. Journal of Building Engineering. 2022 Apr;47:103822.
16. Rombach R, Blattmann A, Lorenz D, Esser P, Ommer B. High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022. p. 10684–95.
17. Shabani MA, Hosseini S, Furukawa Y. HouseDiffusion: Vector Floorplan Generation via a Diffusion Model with Discrete and Continuous Denoising. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) [Internet]. Vancouver, BC, Canada: IEEE; 2023 [cited 2024 May 20]. p. 5466–75. Available from: https://ieeexplore.ieee.org/document/10204517/
18. Stiny G. Introduction to shape and shape grammars. Environment and planning B: planning and design. 1980;7(3):343–51.
19. Wang TC, Liu MY, Zhu JY, Tao A, Kautz J, Catanzaro B. High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition [Internet]. Salt Lake City, UT, USA: IEEE; 2018 [cited 2024 Jun 25]. p. 8798–807. Available from: https://ieeexplore.ieee.org/document/8579015/
20. Weber RE, Mueller C, Reinhart C. Automated floorplan generation in architectural design: A review of methods and applications. Automation in Construction. 2022 Aug;140:104385.
21. Wu W, Fu XM, Tang R, Wang Y, Qi YH, Liu L. Data-driven interior plan generation for residential buildings. ACM Trans Graph. 2019 Dec 31;38(6):1–12.
22. Yi H, Yi YK, Chan T. Performance Based Architectural design optimization: Automated 3D space Layout using simulated annealing. In: Proceedings of the 2014 ASHRAE/IBPSA-USA Building Simulation Conference, Atlanta, GA, USA. 2014. p. 10–2.
23. Zeng P, Gao W, Yin J, Xu P, Lu S. Residential floor plans: Multi-conditional automatic generation using diffusion models. Automation in Construction. 2024 Jun;162:105374.