




FINE-TUNING STABLE DIFFUSION FOR GENERATING 2D FLOOR PLANS USING PROMPT TEMPLATES

Ahmed Mostafa, Omar Amir , Ali M. Mohamed  and Marwa O. Al Enany* 

Higher Institute of Computer Science and Information Systems,

October 6 University, Culture and Science City, Egypt

*Corresponding author: Marwa O. Al Enany (marwaanny33@gmail.com)

Submitted: 02 Feb 2025 Accepted: 20 May 2025 Published: 30 Sep 2025

License: CC BY-NC 4.0 

Abstract Automated generation of 2D floor plans is crucial for architectural design, requiring models to balance precision and adaptability to user-defined specifications. Diffusion models, like Stable Diffusion, excel at generating high-quality images but lack an intrinsic understanding of structured layouts such as floor plans. Conversely, Graph Neural Networks (GNNs) are adept at encoding relational data, representing floor plan objects as nodes and their connections as edges, but they are not generative or capable of processing textual inputs. In this work, we fine-tune Stable Diffusion 1.5 on a custom dataset of floor plans, leveraging structured prompt templates to constrain the model's creativity and guide it toward generating concise, error-tolerant outputs. This research suggests integrating the generative capabilities of diffusion models with the representational strengths of GNNs to overcome inherent challenges in diffusion models, like their inability to explicitly encode spatial relationships. This integration could expand the capabilities of these models, empowering them to comprehend and produce structured layouts more effectively. While computational constraints limited our exploration of this hybrid architecture, our results demonstrate that prompt engineering and dataset preprocessing significantly improve the output quality. This study highlights the potential for generative models in architectural tasks and lays the groundwork for integrating logical reasoning into diffusion-based architectures.

Keywords: graph neural networks (GNNs), diffusion model, latent diffusion, floorplan representation.

1. Introduction

Human culture deeply intertwines with the history of architecture and architectural drawings, reflecting the ability to conceptualize and design living spaces. Architectural drawings, meticulously crafted by hand and archived on paper or as scanned raster images, continued until the second half of the 20th century. Even with the widespread adoption of computer-aided design (CAD) software [7], most architectural drawings remain primarily distributed in image formats, limiting the depth of information that can be extracted and analyzed.

The challenge of generating floor plans that meet specific user requirements has long been a complex task for architects and designers [33]. Traditional methods rely heavily on manual design processes, requiring extensive expertise and time-consuming iterations. This process is inherently complex and requires meticulous attention to detail. Architects and engineers must consider a myriad of factors, including spatial relationships, structural integrity, building codes, and client preferences. To ensure both functionality

and compliance with regulatory standards, architects and engineers must precisely configure each element, from the placement of walls, doors, and windows to the allocation of rooms and incorporation of utilities. Moreover, the manual approach to floor plan creation poses challenges in efficiency and productivity. Engineers must painstakingly adjust dimensions, realign components, and ensure consistency across different sections of the plan. This meticulous work not only prolongs project timelines but also redirects valuable resources towards less innovative design aspects. The pressure to deliver accurate and high-quality floor plans under tight deadlines further exacerbates the strain on professionals in the industry.

In light of these challenges, there is a growing need for automated solutions [35] that can streamline the floor plan generation process. An effective solution would reduce the manual workload, minimize errors, and accelerate the design phase, allowing engineers and architects to focus on creativity and innovation. Automation can also enhance the ability to rapidly explore multiple design alternatives, providing clients with a broader range of options and facilitating more informed decision-making.

Recent advancements in artificial intelligence and machine learning have opened new possibilities for automated floor plan generation [2, 9, 25, 32]. Unlike traditional design methods that rely solely on human expertise, AI-driven approaches can rapidly generate multiple design iterations based on input parameters.

Diverse machine learning methodologies have been employed in floor plan analysis. Convolutional Neural Network-based methodologies have been predominantly utilized due to their applicability to many sorts of floor-plan photographs [5]. CNN-based methodologies necessitate only fundamental image pre-processing techniques and exhibit robustness to floor plan noise. Furthermore, they can be utilized across many drawing styles without necessitating modification, rendering them quick and adaptable.

Nevertheless, due to the pixel-level segmentation employed by these approaches, accurately capturing the precise contours of indoor features is challenging. To address this issue, some methodologies have integrated supplementary post-processing processes that refine the neural network's output. However, this results in the loss of features inherent to the original indoor elements, such as the representation of polygons as line vectors [6, 18]. For instance, walls must possess distinct thickness and area, but, when the shapes become indistinct during the convolution layers, the walls are ultimately represented as line vectors by the post-processing techniques.

For certain user applications, such as representing navigable areas in IndoorGML format [31], abstracting a floor plan layout using machine learning models may be crucial. However, the inherent flexibility and deformability of vector data allows for the adaptation of vector outputs that preserve the original floor plan's form into various objects based on user intent.

This research contributes to developing a novel model that generates 2D floor plans

automatically by leveraging user-specified inputs such as the number of bedrooms, desired room types, and other key architectural constraints. The proposed model aims to address several key challenges in automated floor plan generation:

- translating abstract user requirements into precise spatial configurations,
- ensuring functional and logical room relationships,
- maintaining architectural design principles,
- providing rapid, customizable design solutions.

By utilizing stable diffusion, a generative AI technique, the research demonstrates the potential of AI to streamline the initial stages of architectural design. The model leverages the diffusion model's ability to generate complex, contextually coherent images by progressively denoising latent representations, enabling the creation of floor plans that transform user inputs into detailed spatial layouts. Another contribution is exploring the capabilities and limitations of diffusion models in generating reliable 2D floor plans while proposing hybrid approaches that combine generative and graph-based methods to further push the boundaries of the field.

The remainder of this paper is organized as follows. Section 2 reviews related work in floor plan generation. Section 3 presents our proposed framework. Section 4 describes the dataset design. Section 5 details the Stable Diffusion model structure. Section 6 covers the implementation details. Section 7 presents the evaluation results. Section 8 discusses the findings, and Section 9 concludes the paper.

2. Related work

The generation of 2D floor plans has advanced significantly, leveraging various generative models to address challenges in structured design and adaptability to user-defined parameters. This section highlights key contributions that inform and contextualize this work, focusing on diffusion models, text-conditioned generation, and data-structure-driven approaches.

2.1. Diffusion-based models for floor plan generation

The author in [11] recommend using a diffusion-based approach to create realistic floor-plan images that include room types, furniture specifications, and fenestration details which were often omitted in earlier models, like HouseGAN++ [21]. This method surpasses current results and generates helpful floorplans. However, the direct pixel-input method limits scalability. Two solutions, Cascade Diffusion models and Latent Diffusion models [28], have been introduced to address this issue. Cascade diffusion models incrementally improve image resolution, while Latent Diffusion models map high-dimensional inputs into a more manageable latent space.

The authors in [30] introduced a diffusion-based method that directly predicts a list of polygons for each room, utilizing a transformer architecture. This approach employs a denoising process on 2D coordinates of room and door corners, integrating both discrete and continuous denoising steps to establish geometric relationships such as parallelism and orthogonality. Evaluations on the RPLAN dataset [36, 37] demonstrated significant improvements over state-of-the-art methods, with the capability to generate non-Manhattan structures and control the exact number of corners per room.

A novel approach called HouseCrafter is proposed in [22] that transforms 2D floorplans into complete 3D indoor scenes. By adapting a 2D diffusion model trained on web-scale images, the method generates consistent multi-view RGB-D images across different locations of the scene. These images are generated autoregressively, guided by the floorplan, ensuring consistency and enabling high-quality 3D scene reconstruction. Experiments on the 3D-Front dataset demonstrated the effectiveness of HouseCrafter in generating house-scale 3D scenes.

Further on, in [10] a shear wall layout generation method based on a diffusion process is proposed. Compared with the StructGAN method, the diffusion-based approach demonstrates improved performance in generating realistic and efficient shear wall layouts, contributing to advancements in structural design automation. According to enhancing diffusion models, in [26] the diffusion models in the context of computational design are assessed, particularly on floor plans. A method for refining diffusion models via semantic encoding is suggested. The semantic encoding proposed in this paper enhanced the validity of produced floor plans to 90%. Nevertheless, the article also highlights deficiencies in existing diffusion models, primarily because to an absence of semantic comprehension.

2.2. Text-Conditioned Floor Plan Generation

The research of [17] introduced a pioneering dataset called Tell2Design of over 80 000 floor plans paired with natural language descriptions. This work explored the use of Sequence-to-Sequence models for translating textual input into spatially coherent layouts. By addressing architectural constraints through text-conditioned generation, the authors opened avenues for leveraging natural language as a design interface, making floor plan generation accessible and user-friendly. However, experimental results show that current text-conditional picture generation methodologies fail to address the design creation problem, highlighting the difficulty in understanding ambiguous information and the characteristics of design diversity in the task.

The study in [39] proposed a two-phase method for text-to-floorplan generation, leveraging large language models (LLMs) to create initial layouts from textual descriptions. The approach integrates LLMs to interpret and generate spatial configurations, enhancing the alignment between user requirements and generated designs. The collaboration between LLMs and visual generative models in [8] generates LayoutGPT which is

a method that converts a big language model into a visual planner via in-context learning and CSS style prompts. LayoutGPT can generate credible visual configurations in both image space and three-dimensional indoor environments besides improving image compositions by producing accurate layouts and obtaining performance in interior scene synthesis that is comparable to supervised methods.

Another study in [14] presented a method to automatically render 2D floor plan images from natural language descriptions. This work represents an early attempt to synthesize floor plans directly from textual inputs, bridging the gap between language and visual design in the architectural domain.

2.3. Data Structure-Driven Approaches

For data structure-driven approaches, in [19] a framework that focuses on numerical attributes of floor plans is proposed, including room dimensions and intermediate representations, to ensure adherence to constraints and enhance functional accuracy. New datasets and evaluation metrics are introduced, providing insights into integrating data structures for improved generative modeling of architectural layouts. The study fine-tunes a large language model (LLM), Llama3, but finds it flawed in accurately producing rooms with areas corresponding to computed polygons.

The technique proposed in [1] attempts to present floorplans using numerical vectors that encode design semantics and human behavioral characteristics. The framework comprises two components. The initial component features an automated program that transforms floorplan photos into attributed graphs. The features consist of design semantics and human behavioral characteristics produced by simulation. In the second component, it introduced an innovative LSTM Variational Autoencoder for the purposes of embedding and producing floorplans. The qualitative, quantitative, and expert assessments indicate that this embedding system generates significant and precise vector representations for floorplans, demonstrating its capacity for creating new floorplans.

Authors in [15] developed GenFloor, an interactive design system that generates optimized spatial layouts based on geometrical, topological, and performance constraints. The system introduces novel permutation methods for existing space layout graph representations, such as O-Tree and B*-Tree, enabling the generation of diverse floor plans that meet specified design criteria. GenFloor facilitates designers in their generative design workflow by providing a user-friendly interface and evaluation functionalities.

Research on extracting structured data from image floor plans has also informed generative tasks. A notable study [23] introduced techniques for creating vector and raster representations of floor plans to improve localization accuracy in indoor positioning systems. Though not directly focused on generation, this work highlights the importance of accurate preprocessing and representation, essential for downstream applications. The study proposes a computer vision method for automated map annotation, which significantly reduces the processing time from 40 minutes to 5 minutes. Despite the method's

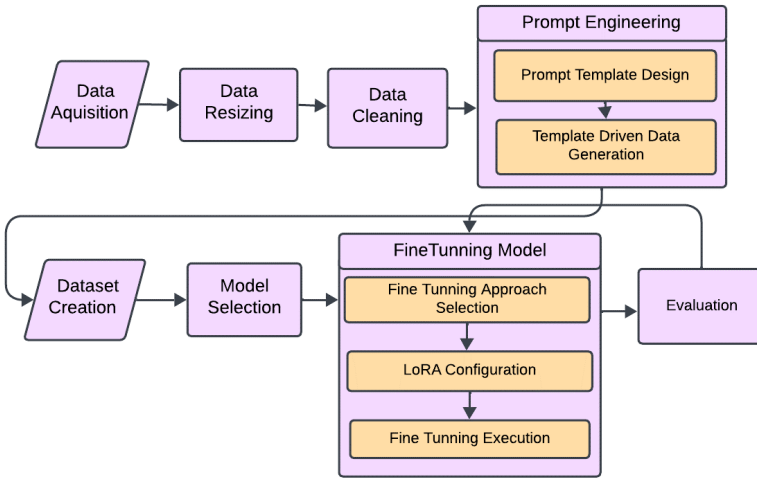


Fig. 1. The stages involved in the process of fine-tuning Stable Diffusion for floor plan generation.

limitations, users can consistently achieve the map model with minimal user modifications.

Most of the previously mentioned studies collectively illustrate the evolving landscape of generative modeling for architectural design. From enhancing realism and accuracy through diffusion-based techniques to leveraging structured prompts and intermediate representations, these advancements underscore the potential for integrating diverse methodologies.

3. The proposed framework

Our proposed framework for fine-tuning Stable Diffusion for floor plan generation consists of several key stages, as illustrated in Figure 1.

3.1. Data acquisition and preparation

This is the initial step in gathering information to train the model. The data used in this study consisted of a curated collection of 300 floorplan images. These images are obtained from various publicly available architectural and design repositories to ensure diversity in design styles and layouts. The primary goal is to create a dataset suitable for training a Stable Diffusion model capable of understanding architectural floorplans.

3.2. Data resizing

Processing the acquired data to match the required dimensions or specifications. To ensure consistency across the dataset and compatibility with the neural network architecture, all floorplan images are resized to 256×256 pixels. This resolution has been selected as it strikes a balance between computational efficiency and preserving sufficient detail in the architectural features. The resizing process employs bicubic interpolation to minimize distortion and preserve the original proportions of the designs.

3.3. Data cleaning

Removing noise, inconsistencies, and irrelevant data from the dataset to ensure high-quality input. The raw floorplan images contained extraneous elements such as annotations, text labels, and other metadata that were not essential for the intended application. To address this, all images underwent a manual and automated cleaning process. This step involves removing textual and graphical artifacts while retaining the structural integrity of the floorplans. Open-source image editing tools and Python-based libraries, such as OpenCV and PIL, are used to streamline the cleaning process.

3.4. Prompt engineering

To enhance task-specific understanding by creating well-structured and targeted inputs. Two steps are included:

- **Prompt Template Design:** Design templates for input prompts to ensure the model understands tasks effectively.
- **Template-Driven Data Generation:** Use the designed templates to generate additional synthetic data or reformat existing data to align with the task requirements.

3.5. Dataset creation

Combining processed and cleaned data to create a final dataset suitable for fine-tuning. This might include integrating real and synthetic data. Following preprocessing, the cleaned and resized images are organized into a structured dataset. Each image is saved in a standardized format (e.g., PNG) to maintain quality and reduce potential issues arising from compression artifacts. An accompanying CSV file stores metadata associated with each image, including source information and preprocessing steps, to improve reproducibility.

3.6. Model selection

Choose the base model to fine-tune. This could involve selecting a pre-trained model that aligns with the task domain.

3.7. Model fine-tuning

- **Fine-Tuning Approach Selection:** Decide on the strategy for fine-tuning (e.g., full model fine-tuning, Low-Rank Adaptation (LoRA), or adapters).
- **LoRA Configuration:** If using LoRA, configure its parameters to efficiently fine-tune large models with minimal resources.
- **Fine-Tuning Execution:** Perform the fine-tuning process using the prepared dataset and configurations.

3.8. Evaluation

Assessing the fine-tuned model's performance against predefined metrics or benchmarks. The results inform whether further adjustments are needed. The process includes feedback loops where insights from the evaluation stage or fine-tuning process may inform modifications in earlier stages, such as dataset creation, prompt design, or model configurations.

4. Dataset designing and characteristics

The main objective is to produce realistic 2D floor plan designs that adhere to a set of linguistic instructions detailing the basic parts of the floor plan. Each data sample consists of a collection of prompts that outline the essential elements of the corresponding floor plan design, which encompass: Semantics that defines the type of the described rooms, Geometry which defines the size and shape of each room, and Topology that illustrates the relationships between various rooms. It can be classified into three categories: relative location, connectedness, and inclusion. The objective is a systematic interior arrangement that conforms to the provided linguistic directives.

- **Volume:** The dataset comprises 313 images, providing a moderately sized collection for training and validation purposes.
- **Diversity:** The dataset encompasses a wide range of architectural styles, including residential, commercial, and mixed-use layouts, ensuring broad applicability of the trained model.
- **Quality Control:** Each image was reviewed post-preprocessing to verify that all unnecessary elements were successfully removed and that the structural details were preserved.

This dataset forms the foundation for the subsequent stages of model training and evaluation, ensuring high-quality inputs and consistency throughout the study. Figure 2 shows samples from our dataset.

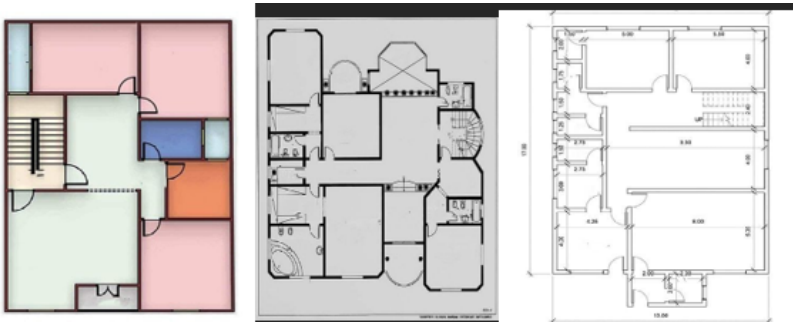


Fig. 2. Samples from the designed dataset showing variety in floor plan designs.

5. Stable Diffusion model structure

Stable Diffusion v1.5 [28, 29] is an advanced text-to-image synthesis model that leverages the principles of diffusion processes within a latent space to generate high-quality images conditioned on textual input. It builds upon the foundational work in diffusion models and latent variable models, integrating them to produce a scalable and efficient framework for image generation tasks [27]. At its core, Stable Diffusion v1.5 is a type of Latent Diffusion Model (LDM) that operates in a compressed, lower-dimensional latent space rather than directly in the high-dimensional pixel space. This approach significantly reduces computational overhead while preserving the ability to generate detailed and coherent images [32]. The decision to utilize Stable Diffusion v1.5 rather than more recent versions such as v2.0 or SDXL was both strategic and deliberate. Version 1.5 is widely regarded for its balance between quality, control, and compatibility with a broad ecosystem of tools and community-created resources. Unlike later models that introduced significant architectural changes—such as a new VAE, deeper prompt sensitivity, and more restrictive output filtering—v1.5 offers a more consistent and interpretable output across a variety of prompts, which is essential for projects requiring reproducibility and detailed prompt engineering. Additionally, the abundance of pretrained models, custom LoRAs, and fine-tuned checkpoints built on v1.5 significantly enhance flexibility and creativity without the computational overhead of retraining. Using v1.5 thus ensures that the work remains accessible, adaptable, and efficient, with outputs that are not only high in visual fidelity but also aligned with the project's specific creative and technical needs [20]. The main architecture of Stable Diffusion v1.5 model consists of the elements described in the following Subsections 5.1, 5.2, 5.3 and 5.4.

5.1. Text encoder

The model utilizes a pre-trained text encoder, typically derived from the CLIP (Contrastive Language-Image Pretraining) architecture developed by OpenAI, which aims to align text and image embeddings in a shared latent space. The CLIP Text Encoder is typically a variant of the Transformer architecture, processing the input text and outputs a fixed-length embedding vector.

The text encoder transforms input textual prompts into [4] a high-dimensional vector representation. It extracts meaningful features from the input text and transforms it into a compact, high-dimensional vector representation, providing semantic guidance to the diffusion model.

The encoded text vector is used as a conditioning input for the U-Net in the diffusion model, which uses cross-attention mechanisms to ensure the generated images align with the textual description. The encoded text vector is pretrained to generalize well across different prompts and concepts, captures nuanced relationships between words, and can work with multiple languages or dialects with fine-tuning.

However, the CLIP Text Encoder may struggle with highly abstract or nonsensical prompts or cultural or domain-specific nuances not covered during training. By leveraging the robust CLIP Text Encoder, Stable Diffusion v1.5 achieves an effective translation of textual descriptions into high-quality images, balancing semantic richness with computational efficiency.

5.2. Latent space representation

Stable Diffusion v1.5 utilizes latent space representation, a compact, high-level mathematical representation of data, to encode and manipulate data in a compressed, lower-dimensional space. This lower-dimensional representation is used in conjunction with a Variational Autoencoder (VAE) and a U-Net architecture to reduce computational complexity and optimize the model's operation. The VAE encoder compresses high-dimensional image data into a latent space and reconstructs it back into pixel space, while the U-Net operates on this latent tensor during the diffusion process. The model learns to denoise latent representations in reverse diffusion steps, optimizing for a perceptual loss to maintain high fidelity to the original data. At inference time, the trained model refines a noisy latent tensor into a meaningful latent representation, which is then reconstructed by the VAE decoder. This approach allows Stable Diffusion to work efficiently with large datasets and generate high-quality images with less computational overhead. Overall, latent space representation is a core component of Stable Diffusion v1.5, enabling efficient processing and high-quality image generation [24].

5.3. Diffusion process

The diffusion model is a U-Net architecture which is neural network responsible for predicting noise at each timestep and is adapted for denoising tasks in the latent space. The forward diffusion process involves gradually adding Gaussian noise to the latent variables over several time steps. This process, also called the noising process, gradually adds Gaussian noise to an input image over a fixed number of steps. On the other hands, the reverse diffusion process entails learning to denoise these latent variables to recover the original data distribution. This process, also called the denoising process, starts from pure noise and attempts to reconstruct the original image. This is where the model learns to “undo” the noise added during forward diffusion. Stable Diffusion performs this process in a latent space, rather than pixel space, using a LDM for efficiency [16,38]. Latent diffusion offers several advantages, including efficiency, scalability, and quality of generated images. By operating in the latent space, the model reduces data dimensionality, resulting in lower computational requirements and faster training times. The compact latent space representation also allows for high-resolution image generation, ensuring high-fidelity, fine-detail images.

5.4. Conditioning mechanism

The model incorporates a conditioning mechanism that integrates the textual embeddings from the text encoder into the diffusion model at each time step. This alignment ensures that the denoising process is guided by the semantic content of the input text, enabling coherent text-to-image synthesis [3]. Stable Diffusion incorporates conditioning to guide the reverse diffusion process toward generating specific outputs. Using a text encoder, textual information is embedded into a high-dimensional space and injected into the U-Net as cross-attention layers, or other Inputs: The process can also be conditioned on images, masks, or other inputs, enabling tasks like inpainting or image-to-image generation.

6. Implementation

To realize the proposed solution of generating precise 2D floor plans using Stable Diffusion v1.5, we implemented a fine-tuning process utilizing Hugging Face’s Low-Rank Adaptation (LoRA) framework [12, 13]. This approach allowed us to adapt the pre-trained diffusion model to our specific task without the need for extensive computational resources or retraining from scratch. Leveraging LoRA, we injected trainable rank decomposition matrices into the attention layers of the Transformer architecture within Stable Diffusion v1.5. This method effectively reduced the number of trainable parameters, making the fine-tuning process more efficient while maintaining the model’s

capacity to learn task-specific representations. The implementation process proceeded as follows:

- The Hugging Face Transformers library was set up, ensuring compatibility with the LoRA integration. The pre-trained Stable Diffusion v1.5 model was loaded as the base model for fine-tuning. We configured the LoRA parameters to insert low-rank adaptation matrices into the attention layers, specifying the desired rank to balance between computational efficiency and model expressiveness.
- The fine-tuning dataset, comprising pairs of constrained prompts and corresponding floor plan images, was prepared to align with the requirements of LoRA training. Each prompt is structured consistently, varying only in numerical parameters, as previously described. Although the details of data preparation are discussed elsewhere, it is important to note that the dataset was formatted to be compatible with the Hugging Face dataset utilities, enabling seamless integration into the training pipeline.
- During the training process, the standard optimization techniques were utilized. The optimizer was set to AdamW with a learning rate carefully selected to ensure stable convergence without overfitting. We adopted a learning rate scheduler to adjust the learning rate dynamically based on the training progress.
- The loss function was configured to emphasize the reconstruction accuracy of the floor plans. While the primary objective remained the minimization of the denoising score matching loss inherent in diffusion models, we integrated additional components to focus on the structural aspects of the floor plans. Specifically, edge-aware loss functions that penalized discrepancies in the line structures between the generated and ground truth images was incorporated. This helped the model prioritize the preservation of architectural details crucial for floor plan accuracy.
- Training was conducted on hardware equipped with GPUs capable of handling the computational demands of the model. The use of LoRA significantly reduced memory usage, allowing the fine-tuning process to be executed on standard GPU setups without requiring distributed training or specialized hardware.
- The training progress was monitored by evaluating intermediate outputs and loss convergence. Visual inspections of generated floor plans were performed to ensure that the model was learning to produce outputs that adhered to the specified parameters in the prompts. Any signs of the model reverting to overly creative outputs were addressed by adjusting training hyperparameters, such as the learning rate or weight decay.
- Upon completion of the fine-tuning, the adapted model was saved using Hugging Face's model saving utilities. This enabled easy deployment and sharing of the model for inference tasks.

The final model was capable of generating precise 2D floor plans that accurately reflected the numerical specifications in input prompts.

7. Evaluation and Results

To validate the effectiveness of this implementation, evaluations are conducted using a set of test prompts with varying parameters. The generated floor plans are assessed for accuracy in room counts, spatial arrangements, and adherence to architectural conventions.

The performance of the fine-tuned Stable Diffusion model in generating accurate 2D floor plans is done by employing the Structural Similarity Index Measure (SSIM) as an evaluation metric. The initial proposal for the Structural Similarity Index was made in [34] by Wang, Bovik et al. in 2004. The two images being compared must be appropriately sized and aligned in order to compare them point by point. A sliding $N \times N$ (usually 11×11) Gaussian weighted window is used for the computations. Luminance, contrast, and structure are the three similarity functions that are computed on the windowed image data. The general form of the SSIM index is then created by combining the three mentioned similarity functions as:

$$\text{SSIM}(x, y) = [l(x, y)] \cdot [c(x, y)] \cdot [s(x, y)] \quad (1)$$

where l , c , and s compare luminance, contrast and structure, respectively.

The ability of this index to mimic human subjectivity is its strongest attractive point. Specifically, changes in the spatial arrangement of image brightness have a significant impact on the Human Visual System (HVS) and the SSIM Index. SSIM is particularly suited for this implementation as it assesses the structural resemblance between two images, focusing on spatial configurations and line structures essential in architectural designs. The evaluation involves generating floor plan images based on prompts from a test set using both the base Stable Diffusion model and our fine-tuned model. Each generated image was compared to its corresponding ground truth floor plan using SSIM.

For implementation and testing purposes, the Fine-tuned model was tested with two different prompts. The first prompt was unusual or rarely spread in design as

- **Prompt 1:** “floor plan of house having two living rooms, one bedrooms, three bathroom, two kitchen, one garage, three store, one entrance.”

The SSIM Scores of Diffusion model versus fine-tuned Diffusion model for the mentioned prompt are summarized in Table 1, numbers 1 and 2, while 10 generated images from the fine tuned Stable Diffusion model are presented in Figure 3.

The SSIM index generally ranges from -1 to 1 , with 1 signifying perfect similarity, 0 denoting no similarity, and -1 representing perfect anti-correlation. Our fine-tuned model achieved an average SSIM score of 0.3426 , surpassing the base model’s average SSIM score of 0.3191 . The higher SSIM score indicates that the fine-tuned model produces floor plans that are structurally more similar to the ground truth images, demonstrating enhanced accuracy in capturing the architectural details specified in the prompts. The results demonstrated that this fine-tuned model successfully generated

Tab. 1. SSIM scores for base Diffusion model and fine-tuned Diffusion model.

No.	Model	SSIM score
1	Base Model	0.3191
2	Fine-Tuned Model (Prompt 1)	0.3426
3	Fine-Tuned Model (Prompt 2)	0.4254

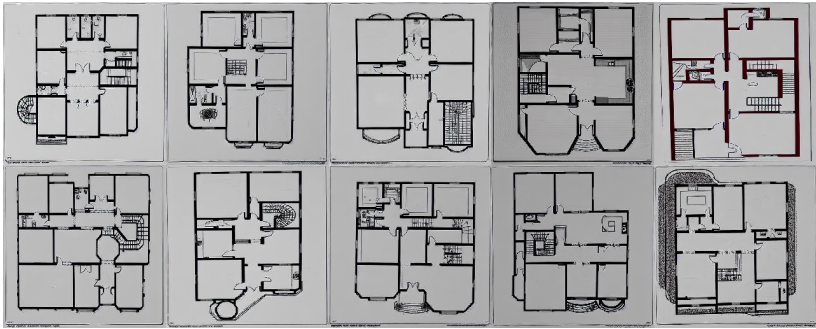


Fig. 3. Generated images from Prompt 1: “floor plan of house having two living rooms, one bedrooms, three bathroom, two kitchen, one garage, three store, one entrance.”

floor plans that met the specified criteria, confirming the efficacy of our implementation strategy using Hugging Face’s LoRA framework.

The second tested prompt was more popular and familiar in most home designs:

- **Prompt 2:** “floor plan of house having one living room, two bedrooms, one bathroom, one kitchen, one hall, one entrance.”

The proposed model has generated 10 designs that matches the mentioned prompt and give more varieties for the designer from this prompt. The SSIM Scores of Diffusion model versus fine-tuned Diffusion model for the mentioned prompt are summarized in Table 1, number 3, while 10 generated images from the fine tuned Stable Diffusion model are presented in Figure 4.

The improvement can be attributed to this approach of constraining the input prompts during fine-tuning. By limiting prompts to a static structure with only key numerical parameters varying—such as the number of bedrooms or dining rooms—we guided the model to focus on these critical elements. This constraint reduces unnecessary creativity, enabling the model to generate floor plans that more precisely reflect the specified requirements.

Although the numerical increase in SSIM is modest, it signifies meaningful enhancements in the context of architectural design, where precision is paramount. Even small



Fig. 4. Generated images from Prompt 2: “floor plan of house having one living room, two bedrooms, one bathroom, one kitchen, one hall, one entrance.”

improvements in SSIM correspond to better alignment of walls, rooms, and spatial relationships, which are crucial for the practical usability of floor plans. The fine-tuned model’s outputs exhibit greater fidelity to the intended layouts, suggesting that our method effectively bridges the gap between creative image generation and the need for exactitude in architectural applications. These results validate that constraining the model’s input prompts and fine-tuning it on specialized data can improve its performance in generating accurate floor plans. The fine-tuned model demonstrates a better understanding of the correlation between the specified numerical parameters and the spatial configurations required, making it a more reliable tool for architectural design tasks that demand high levels of precision.

8. Discussion

To assess the performance of Stable Diffusion v1.5 in producing architectural design outputs, two floor plan prompts were evaluated using expert judgment based on design clarity, feature correctness, and alignment with the prompt. A panel of 3 evaluators with experience in architecture, interior design, and AI image generation rated the outputs using the following criteria:

- **Relevance to Prompt:** Correct inclusion and quantity of specified rooms and features.
- **Layout Clarity:** Logical and readable floor plan arrangement.
- **Design Aesthetics:** Overall visual structure and neatness.
- **Spatial Realism:** Realistic spatial proportions and plausible connectivity between rooms.
- **Prompt Sensitivity:** Ability of the model to reflect prompt changes between two variants.

Tab. 2. Expert Evaluation of Generated Floor Plans

Prompt	Relevance	Clarity	Aesthetics	Spatial Realism	Average
Prompt 1	4	3	4	3	3.8
Prompt 2	5	4	4	4	4.4

Table 2 displays the results.

Prompt 1

The model captured most of the required rooms but struggled with accurate quantity differentiation, especially for “three stores” and “two kitchens”, which were either combined or misrepresented. The overall spatial layout lacked architectural realism (e.g., bathrooms sometimes placed without adjacent bedrooms or hall access), although room labeling was reasonably intuitive.

Prompt 2

The model performed better with this simpler prompt. All rooms were represented clearly, and the spatial layout was more plausible and visually coherent. Room positioning followed a logical flow, and the floor plan adhered closely to modern residential design principles. The evaluation indicates that Stable Diffusion v1.5 is effective for generating conceptual and visually descriptive floor plans from textual prompts, especially when the prompts are concise and moderately complex. However, for prompts with a high number of room types or quantities, the model’s spatial reasoning and ability to differentiate repeating elements (e.g., multiple stores or kitchens) become more limited. These findings suggest the model is well-suited for early-stage ideation and visual storytelling, but not for technical architectural planning.

9. Conclusion

This study demonstrated the effective fine-tuning of Stable Diffusion v1.5 for the generation of precise 2D floor plans. By constraining the model with structured prompts that varied only in specific numerical parameters, we guided it to focus on accuracy and the nuanced spatial configurations essential in architectural designs. This approach successfully reduced unnecessary creativity inherent in diffusion models, resulting in outputs that more closely adhered to the specified requirements. The improved performance, reflected in higher SSIM scores compared to the base model, highlights the potential of combining prompt engineering with fine-tuning to adapt generative models for tasks demanding exactitude. Our findings indicate that diffusion models can be tailored to produce functionally accurate and detailed images in domains where precision is

paramount, expanding their applicability beyond creative image synthesis to practical, precision-oriented applications.

Acknowledgement

This work was supported by the Higher Institute of Computer Science and Information Systems, Culture and Science City, October 6 University, Egypt.

References

- [1] V. Azizi, M. Usman, H. Zhou, P. Faloutsos, and M. Kapadia. Graph-based generative representation learning of semantically and behaviorally augmented floorplans. *The Visual Computer* 38(8):2785–2800, 2022. doi:[10.1007/s00371-021-02155-w](https://doi.org/10.1007/s00371-021-02155-w).
- [2] S. K. Baduge, S. Thilakarathna, J. S. Perera, M. Arashpour, P. Sharafi, et al. Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications. *Automation in Construction* 141:104440, 2022. doi:[10.1016/j.autcon.2022.104440](https://doi.org/10.1016/j.autcon.2022.104440).
- [3] T. Berrada, P. Astolfi, M. Hall, R. Askari-Hemmat, Y. Benchetrit, et al. On improved conditioning mechanisms and pre-training strategies for diffusion models. In: *Advances in Neural Information Processing Systems*, vol. 37, pp. 13321–13348. Curran Associates, Inc., 2024. https://proceedings.neurips.cc/paper_files/paper/2024/hash/18023809c155d6bbcd27e443043cdebf-Abstract-Conference.html.
- [4] D. Bolya and J. Hoffman. Token merging for fast stable diffusion. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 4599–4603, 2023. doi:[10.1109/CVPRW59228.2023.00484](https://doi.org/10.1109/CVPRW59228.2023.00484).
- [5] L.-P. de las Heras, S. Ahmed, M. Liwicki, E. Valveny, and G. Sánchez. Statistical segmentation and structural recognition for floor plan interpretation: Notation invariant structural element recognition. *International Journal on Document Analysis and Recognition (IJDAR)* 17(3):221–237, 2014. doi:[10.1007/s10032-013-0215-2](https://doi.org/10.1007/s10032-013-0215-2).
- [6] S. Dodge, J. Xu, and B. Stenger. Parsing floor plan images. In: *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pp. 358–361, 2017. doi:[10.23919/MVA.2017.7986875](https://doi.org/10.23919/MVA.2017.7986875).
- [7] J. Encarnação, R. Lindner, and E. G. Schlechtendahl. *Computer Aided Design: Fundamentals and System Architectures*. 2nd edn. Springer-Verlag, Berlin, Heidelberg, 2012. doi:[10.1007/978-3-642-84054-8](https://doi.org/10.1007/978-3-642-84054-8).
- [8] W. Feng, W. Zhu, T.-J. Fu, V. Jampani, A. Akula, et al. LayoutGPT: Compositional visual planning and generation with large language models. In: *Advances in Neural Information Processing Systems*, vol. 36, pp. 18225–18250. Curran Associates, Inc., 2023. https://proceedings.neurips.cc/paper_files/paper/2023/hash/3a7f9e485845dac27423375c934cb4db-Abstract.html.
- [9] G. Goodman. *A machine learning approach to artificial floorplan generation*. Master’s thesis, University of Kentucky, 2019. https://uknowledge.uky.edu/cs_etds/89.
- [10] Y. Gu, Y. Huang, W. Liao, and X. Lu. Intelligent design of shear wall layout based on diffusion models. *Computer-Aided Civil and Infrastructure Engineering* 39(23):3610–3625, 2024. doi:[10.1111/mice.13236](https://doi.org/10.1111/mice.13236).
- [11] L. Hahkio. *Generation of realistic floorplans using diffusion-based models*. Master’s thesis, University of Helsinki, 2023. <https://urn.fi/URN:NBN:fi:aalto-202310156363>.

- [12] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, et al. LoRA. Hugging Face. <https://huggingface.co/docs/diffusers/main/en/training/lora>.
- [13] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, et al. LoRA: Low-rank adaptation of large language models. arXiv, arXiv.2106.09685, 2021. doi:10.48550/arXiv.2106.09685.
- [14] M. Jain, A. Sanyal, S. Goyal, C. Chattopadhyay, and G. Bhatnagar. Automatic rendering of building floor plan images from textual descriptions in English. arXiv, arXiv.1811.11938, 2018. doi:10.48550/arXiv.1811.11938.
- [15] M. Keshavarzi and M. Rahmani-Asl. GenFloor: Interactive generative space layout system via encoded tree graphs. *Frontiers of Architectural Research* 10(4):771–786, 2021. doi:10.1016/j.foar.2021.07.003.
- [16] C. Kupferschmidt, A. D. Binns, K. L. Kupferschmidt, and G. W. Taylor. Stable rivers: A case study in the application of text-to-image generative models for Earth sciences. *Earth Surface Processes and Landforms* 49(13):4213–4232, 2024. doi:10.1002/esp.5961.
- [17] S. Leng, Y. Zhou, M. H. Dupty, W. S. Lee, S. C. Joyce, et al. Tell2Design: A dataset for language-guided floor plan generation. arXiv, arXiv.2311.15941, 2023. doi:10.48550/arXiv.2311.15941.
- [18] C. Liu, J. Wu, P. Kohli, and Y. Furukawa. Raster-to-vector: Revisiting floorplan transformation. In: *Proc. 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2214–2222, 2017. doi:10.1109/ICCV.2017.241.
- [19] Z. H. Luo, L. Lara, G. Y. Luo, F. Golemo, C. Beckham, et al. DStruct2Design: Data and benchmarks for data structure driven generative floor plan design. arXiv, arXiv.2407.15723, 2024. doi:10.48550/arXiv.2407.15723.
- [20] Z. Ma, Y. Zhang, G. Jia, L. Zhao, Y. Ma, et al. Efficient diffusion models: A comprehensive survey from principles to practices. arXiv, arXiv.2410.11795, 2024. doi:10.48550/arXiv.2410.11795.
- [21] N. Nauata, S. Hosseini, K.-H. Chang, H. Chu, C.-Y. Cheng, et al. House-GAN++: Generative adversarial layout refinement networks. arXiv, arXiv.2103.02574, 2021. doi:10.48550/arXiv.2103.02574.
- [22] H. T. Nguyen, Y. Chen, V. Voleti, V. Jampani, and H. Jiang. HouseCrafter: Lifting floorplans to 3D scenes with 2D diffusion model. arXiv, arXiv.2406.20077, 2024. doi:10.48550/arXiv.2406.20077.
- [23] M. Opiela, M. Hrehová, and F. Galčík. Map model extraction from image floor plans. In: *Proceedings of the Work-in-Progress Papers at the 13th International Conference on Indoor Positioning and Indoor Navigation (IPIN-WiP 2023)*, vol. 3581 of *CEUR-WS.org: IAOA Series*. Nuremberg, Germany, 2023. https://ceur-ws.org/Vol-3581/194_WiP.pdf.
- [24] L. Papa, L. Faiella, L. Corvitto, L. Maiano, and I. Amerini. On the use of stable diffusion for creating realistic faces: from generation to detection. In: *2023 11th International Workshop on Biometrics and Forensics (IWBF)*, pp. 1–6, 2023. doi:10.1109/IWBF57495.2023.10156981.
- [25] P. N. Pizarro, N. Hitschfeld, I. Sipiran, and J. M. Saavedra. Automatic floor plan analysis and recognition. *Automation in Construction* 140:104348, 2022. doi:10.1016/j.autcon.2022.104348.
- [26] J. Ploennigs and M. Berger. Automating computational design with generative AI. arXiv, arXiv.2307.02511, 2024. doi:10.48550/arXiv.2307.02511.
- [27] A. Razzhigaev, A. Shakhmatov, A. Maltseva, V. Arkhipkin, I. Pavlov, et al. Kandinsky: an improved text-to-image synthesis with image prior and latent diffusion. arXiv, arXiv.2310.03502, 2023. doi:10.48550/arXiv.2310.03502.
- [28] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In: *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 10684–10695, 2022. doi:10.1109/CVPR52688.2022.01042.

- [29] R. Rombach and P. Esser. SD v1.5. Hugging Face, 2024. <https://huggingface.co/stable-diffusion-v1-5>.
- [30] M. A. Shabani, S. Hosseini, and Y. Furukawa. HouseDiffusion: Vector floorplan generation via a diffusion model with discrete and continuous denoising. In: *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023)*, pp. 5466–5475, 2023. doi:10.1109/CVPR52729.2023.00529.
- [31] S. Srivastava, N. Maheshwari, and K. S. Rajan. Towards generating semantically-rich IndoorGML data from architectural plans. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 42:591–595, 2018. doi:10.5194/isprs-archives-XLII-4-591-2018.
- [32] L. Wang, J. Liu, Y. Zeng, G. Cheng, H. Hu, et al. Automated building layout generation using deep learning and graph algorithms. *Automation in Construction* 154:105036, 2023. doi:10.1016/j.autcon.2023.105036.
- [33] X.-Y. Wang, Y. Yang, and K. Zhang. Customization and generation of floor plans based on graph transformations. *Automation in Construction* 94:405–416, 2018. doi:10.1016/j.autcon.2018.07.017.
- [34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13(4):600–612, 2004. doi:10.1109/TIP.2003.819861.
- [35] R. E. Weber, C. Mueller, and C. Reinhart. Automated floorplan generation in architectural design: A review of methods and applications. *Automation in Construction* 140:104385, 2022. doi:10.1016/j.autcon.2022.104385.
- [36] W. Wu, L. Fan, L. Liu, and P. Wonka. MIQP-based layout design for building interiors. *ACM Transactions on Graphics (SIGGRAPH Asia)* 38(6):1–12, 2019.
- [37] W. Wu, X.-M. Fu, R. Tang, Y. Wang, Y.-H. Qi, et al. Data-driven interior plan generation for residential buildings. Wenming Wu's Homepage, 2019. <https://wutomwu.github.io/particulars.html?id=1>. RPLAN project page.
- [38] J. Yang, Z. Cheng, Y. Duan, P. Ji, and H. Li. ConsistNet: Enforcing 3D consistency for multi-view images diffusion. In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7079–7088, 2024. doi:10.1109/CVPR52733.2024.00676.
- [39] Z. Zong, Z. Zhan, and G. Tan. HouseLLM: LLM-assisted two-phase text-to-floorplan generation. arXiv, arXiv.2411.12279v3, 2024. doi:10.48550/arXiv.2411.12279.

