IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Novel House GAN

[1]Prafulkumar Ponnappan, [2]Rahul Rajesh, [3]Rishit Kurup, [4]Suyog Yadav

[1,2,3,4]Student
[1]Department of Computer Engineering,
[1]Pillai College of Engineering (Autonomous), Navi Mumbai, India

Abstract: This paper presents a novel approach for generating floor plans using machine learning techniques, specifically Generative Adversarial Networks (GANs). The implemented method utilizes a bubble graph, which is a rough layout generated by an architect, as input for the GANs. The GANs process the bubble graph and quickly iterate multiple floor plan solutions based on user-specified constraints such as the number of rooms, room sizes, and spatial layout. The generated plans are evaluated against a set of quality metrics such as plan feasibility and aesthetics. The results show that the proposed method is able to generate multiple high-quality floor plans that are comparable to those designed by human experts. Additionally, the approach can significantly reduce the time and cost associated with the traditional floor plan design process. The customer and architect can select a suitable layout that will be converted into a 2D floorplan by the GAN. The implications of this research are significant for the fields of architecture and building design, as it has the potential to revolutionize the way in which floor plans are generated in the future.

***IndexTerms* – Floor Plans, Architecture, Generative Adversarial Network.**

INTRODUCTION

The design of a home is a complex and intricate task that requires a balance between functionality and aesthetics. Floor plans are diagrammatic representation of a room that composes a building and their spatial relationships to one another, play a crucial role in this process. However, creating floor plans can be a strenuous task that requires both imagination and mathematical skills. The process can be further complicated by the need to follow explicit and implicit rules, while also allowing for a high degree of freedom in how these rules are followed.

Home design is not just a professional endeavour, but also a personal one. Many non-architects and non-designers are looking to remodel or build their homes, but hiring private designers can be quite expensive. There is significant value in democratizing the skill of home design through software automation. A software that can design homes by itself or with human input would enable non-professionals to access architectural services that are otherwise too costly. Moreover, such a software would also lower the expenses of the iterative process that is common in professional settings.

This research paper aims to explore the use of machine learning techniques, especially Generative Adversarial Networks (GANs), to automate the process of floor plan generation. The paper will discuss the challenges of floor plan design and how GANs can be used to learn the explicit and implicit rules of floor plan design from existing examples. The proposed system, House-GAN, will be evaluated against metrics such as plan feasibility and aesthetics. The implications of this research are significant for the fields of architecture and building design, as it has the potential to revolutionize the way in which floor plans are generated in the future.

METHODOLOGY

Overview

The proposed system, House-GAN, is a novel approach to automate the process of floor plan generation using machine learning techniques, specifically Generative Adversarial Networks (GANs). The system utilizes a relational generator and discriminator, wherein the input graph constraint is encoded into the graph structure of the relational networks, allowing the system to learn the explicit and implicit rules of floor plan design from existing examples. The system uses Conv-MPN, which differs from traditional GCNs, to take in a noise vector per room and a bubble diagram as input, then generates a house layout as a rectangle aligned to an axis; per room, satisfying the rules of spatial relationships between rooms. The generated floor plan is represented as a graph, where each node represents a specific room with a specific room type and each edge represents the spatial vicinity between rooms. The proposed system aims to improve the efficiency of floor plan generation and learn and implement novel and upcoming technologies in the field of machine learning.

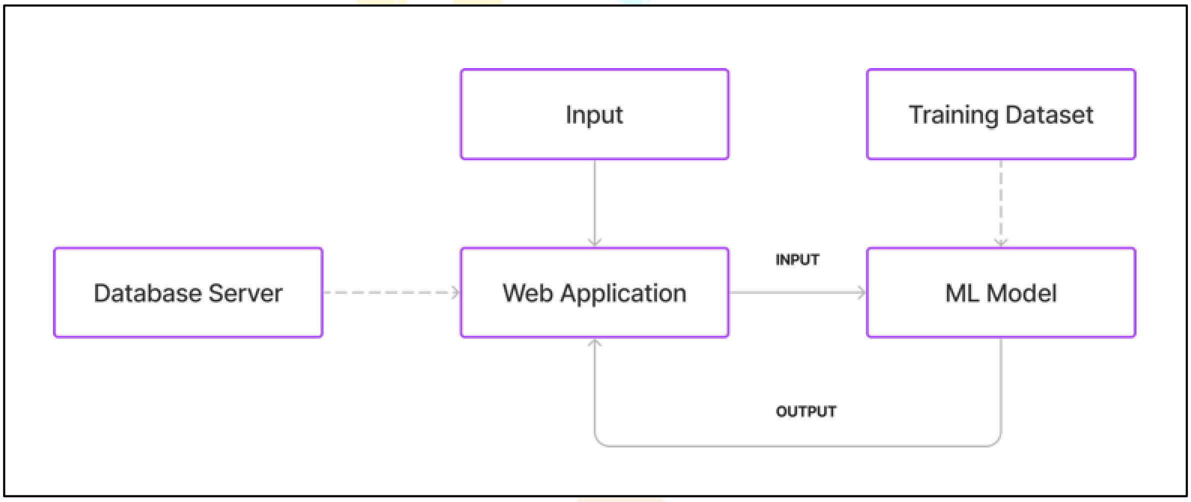
Existing System Architecture

Current floor plan generation systems have limitations such as inefficiency and lack of ability to generate multiple variations of a given plan. These systems are restrictive for both individual users and organizations that employ them, as their primary focus is on providing assistance rather than automating the process. Additionally, the current systems are highly iterative, requiring constant feedback from clients and revisions from architects which can be time-consuming and inhibit the ability to scale and automate the process.

Current floor plan generation systems are often highly iterative, requiring constant feedback and revisions from clients and architects, which can be time-consuming and impede the ability to scale and automate the process. An example of such a system is AutoCAD, which is widely used for creating new designs and floor plans, but does not have the capability to iteratively generate multiple designs and relies on the user to perform the majority of the task. These systems are limited to providing assistance in generating a single plan and rendering the final output, leaving the task of generating multiple floor plans iteratively to the user.

System Architecture

The system will operate as a web-based user interface, where users can upload a bubble graph and input specific constraints for the proposed floor plan to be generated. The system utilizes a combination of Generative Adversarial Networks (GANs) and Conv-MPN, which differs from traditional GCNs, to convert the input bubble graph and constraints into a 2D floor plan as output. The system allows users to efficiently iterate through multiple variations of a required floor plan and gives the flexibility to select the final output. Additionally, the system has the capability to convert the final selected 2D output into a 3D floor plan using Pix2pix, which helps users to visualize and understand the final output in a more intuitive way.



block diagram of system architecture

Machine Learning Model

This project uses a machine learning technique called Generative Adversarial Neural Network (GAN) to create floor plans from bubble graphs. A bubble graph is a sketch made by an architect that shows the rough arrangement of rooms and spaces. The GAN has two parts: one that makes floor plans from bubble graphs (generator), and one that tells if the floor plans are real or fake (discriminator). The generator tries to make floor plans that look good and meet the user's needs, while the discriminator tries to catch any mistakes or flaws in them. The two parts challenge each other until they reach a balance where the generator can trick the discriminator. This way, the GAN can make many different floor plans that are similar in quality to those made by human experts.

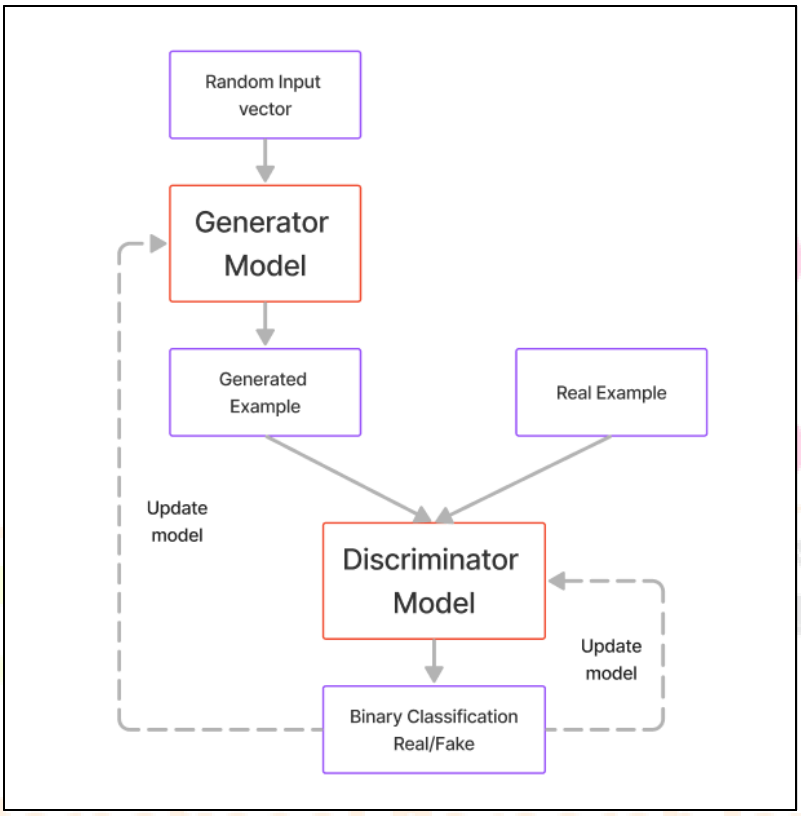
The ML model will be trained on a sample layout dataset; the ML model will have the ability to accept inputs from the user by the means of the web UI that will act as the front end of the implemented system. The ML model will generate suitable outputs based on the input parameters and constraints provided by the user the subsequent output can be viewed on the web application.

GAN

In June 2014, Ian Goodfellow and his colleagues proposed a new type of machine learning framework called generative adversarial network (GAN). This framework involves two neural networks that compete with each other in a game-like scenario, where one network tries to create realistic data and the other network tries to detect fake data. The goal is to improve both networks' performance through this competition.

In this paper, we explore the use of GANs, a deep learning framework that can learn to generate new data that resembles a given dataset. GANs consist of two neural networks that compete with each other in a zero-sum game: a generator that tries to create realistic data and a discriminator that tries to distinguish real data from fake data. The generator is not directly trained to match a specific data point, but rather to deceive the discriminator. This allows the model to learn without supervision. GANs have also been extended to other learning tasks such as semi-supervised learning, supervised learning, and reinforcement learning. GANs can be seen as a form of evolutionary mimicry between the two networks.

The proposed system consists of a ML model which is comprised of a generator and a discriminator. The generator within the model is trained on the sample layout training data and the discriminator already has the sample layout training data which is used for comparison. The generator based on its training data tries to generate layouts which are sent to the discriminator; the discriminator compares the received input to the 'real' sample and the discriminator rejects these inputs until and unless the generator creates a sample that 'fools' the discriminator. Thus, the generator keeps on creating new samples and the discriminator tries to classify them into real and fake samples.



GAN block diagram

Conv-MPN

Convolutional Message Passing Networks (Conv-MPN) is a novel approach to Message Passing Neural Networks (MPNNs) in the field of machine learning. It is a type of Graph Neural Networks (GNNs) that takes a graph as input with node and edge features and computes a function that is depended on both the features and the graph structure. MPNNs, also known as message passing GNNs, update node attributes by sharing data with neighboring nodes. A common MPNN structure has several layers of propagation, where the features of each node are changed by combining the attributes of its neighbors. Conv-MPN is different from traditional GCNs, it uses convolution operation to propagate information between nodes, allowing for more efficient and effective feature aggregation. This approach helps to learn more expressive representations of the graph structure, thus making it useful for tasks such as floor plan generation.

Aggregation functions are parametric in nature, and have three different “flavors” of graph neural networks:

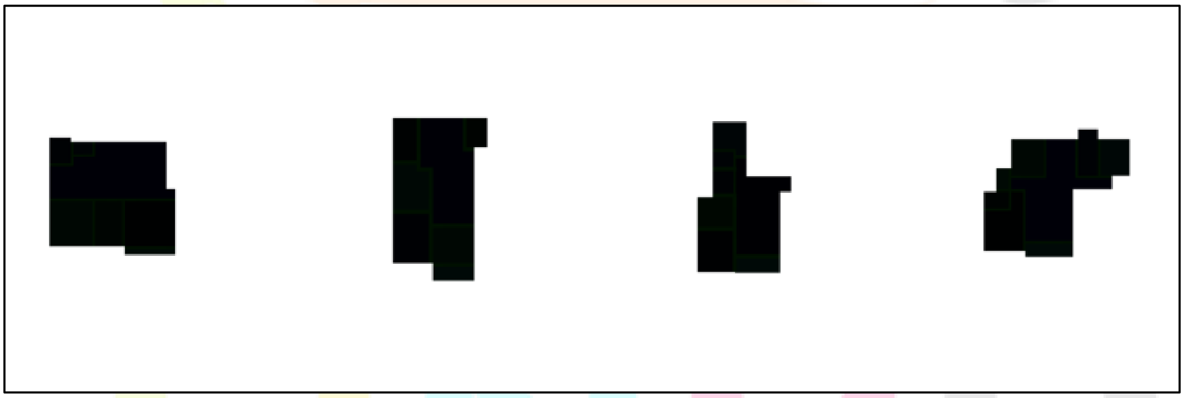
- **convolutional:** linear combination of neighboring features where weights depend only on the structure of the graph.
- **attentional:** linear combination, where weights are computed based on the features.
- **message passing:** a general nonlinear function dependent on the features of two nodes sharing an edge.

Of these three flavors, the latter is the most general and the two others can be seen as special cases of message passing.

Conv-MPN uses a unique approach to update the feature of nodes in the graph. Instead of using two CNN modules as in traditional methods, it uses a pooling operation to gather information from all neighboring nodes to encode a message and then applies a CNN to update the feature vector. This approach reduces the GPU memory requirements and allows for more efficient feature updates. Additionally, it reduces the need for encoding a message for every pair of nodes, which makes the feature updates more effective and expressive.

Training Dataset

The training dataset for the system is extensive and diverse, consisting of over 10,000 images in PNG format. These images represent different floor plan layouts and serve as the basis for training the machine learning model. The generator uses this data to "learn" the relationships and patterns between the different components of a floor plan and generate new samples based on this learning. The discriminator also utilizes these images to validate the output generated by the generator and determine its usability. The robustness and accuracy of the model is heavily dependent on the quality and diversity of this training data, making it a critical component of the system.



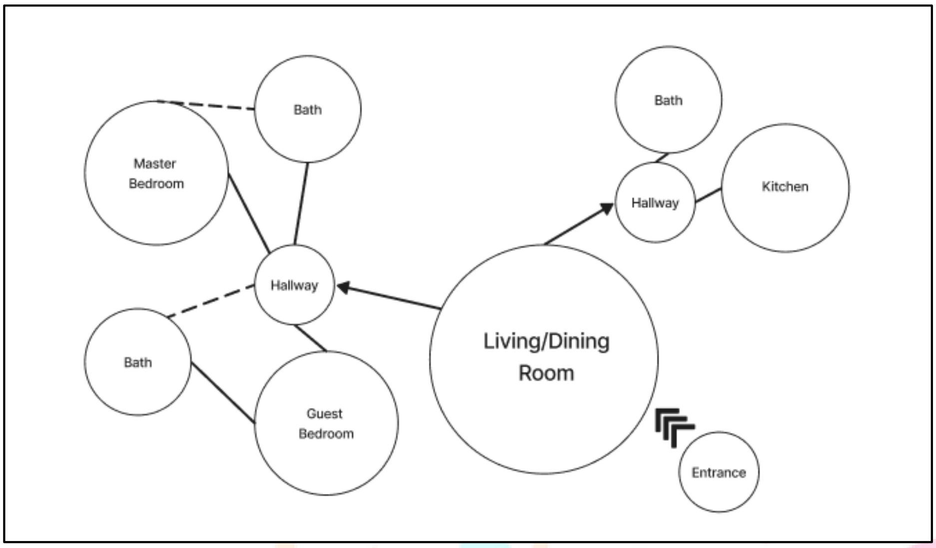
examples of PNG format images used in sample dataset

The input set used in our project consists of:

- Openings
- Footprint
- No. of Rooms
- Bubble Diagram

Bubble Graph

A bubble graph is a type of diagram or chart that represents data points as bubbles, where the size of the bubble represents the magnitude of the data. The position of the bubbles on the graph is used to represent relationships or patterns between different data points. The bubble graph is the input for the Message Passing Neural Network (MPN). It represents the arrangement and connections of various rooms. The MPN facilitates the transfer of this input information to the machine learning model for further processing.

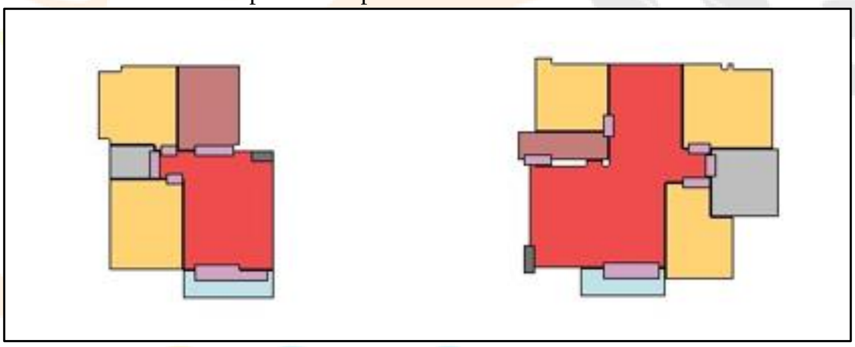


example of bubble graph

Output post model training

The output of the above system is a 2D floor plan generated from user inputted bubble graph and constraints. This generation will be achieved through the use of a GAN and Conv-MPN algorithm. The expected output will consist of multiple two-dimensional floor plan iterations which will be generated based on the bubble graph and the structural constraints provided by the user

expected output from the GAN model



Conclusion

This floor plan generation system is being envisioned with the aim of addressing the limitations of current floor plan generation systems and even working in conjunction with them. Our goal is to create a system that can accept user input through a simple and intuitive web-app, securely store the input in a database, and analyze it using Generative Adversarial Networks. Our implemented system is intended to empower the user while improving and complementing an organization's existing systems, reducing operational costs, increasing user engagement, and enabling faster iteration of plans. The use of a web-app would allow the system to work on any device or operating system that has web access. Our project aims to apply the knowledge gained through our curriculum and research on this topic. Although the project has not been fully implemented yet, we are excited to build a technology demonstrator that showcases the technology behind GAN based on the research compiled in this paper. We anticipate that this technology will be a catalyst for increased automation within the field of design and bring greater accessibility to the process for everyone.

ACKNOWLEDGMENT

Pillai College of Engineering (Autonomous), Navi Mumbai is well equipped with digital resources and standard repositories such as IEEE Xplore, K-Hub, etc; which proved to be incredibly beneficial for this paper. The authors would like to thank Prof. Sheetal P Gawande, our project guide, the Department of Computer Engineering and our Head of Department Dr. Sharvari Govilkar and our Principal Dr. Sandeep Joshi.

REFERENCES

[1] Y. LeCun and Y. Bengio, “Convolutional networks for images, speech, and time series,” in The Handbook of Brain Theory and Neural Networks, 2nd ed., M. A. Arbib, Ed., MIT Press, 2003, pp. 255-258.
[2] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” arXiv preprint arXiv:1411.1784, 2014.
[3] I. J. Goodfellow et al., “Generative adversarial networks,” in Advances in Neural Information Processing Systems 27, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 2672-2680.
[4] X. Chen et al., “InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets,” in Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett, Eds., Curran Associates, Inc., 2016, pp. 2172-2180.
[5] C. Liu*,* J. Wu*,* P. Kohli*,* and Y. Furukawa, “Conv-MPN: Convolutional message passing neural network for structured outdoor architecture reconstruction,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 962-971.
[6] J. Kang and S. Cho, “A machine learning approach to artificial floorplan generation,” Proceedings of the 24th International Conference on Intelligent User Interfaces, 2019, pp. 379-389.
[7] N. Nauata, K. Chang, G. Mori and Y. Furukawa, “House-GAN: Relational generative adversarial networks for graph-constrained house layout generation,” European Conference on Computer Vision, Springer, 2020, pp. 169-185.
[8] S. Khanzadeh Mollaei and A. H. Gandomi, “Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications,” Automation in Construction, vol. 120, 2020, p. 103388.
[9] S. Sarwar and M. R. Islam, “Generating synthetic medical images by using GAN to improve CNN performance in skin cancer classification,” 2020 IEEE International Conference on Informatics, IoT*,* and Enabling Technologies (ICIoT), IEEE*,* 2020, pp. 1-6.
[10] A. Khan, A. Sohail, U. Zahoora and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” Artificial Intelligence Review, vol. 53, no. 8, pp. 5455-5516, Aug. 2020.
[11] M. Bronstein, “Beyond Message Passing, a Physics-Inspired Paradigm for Graph Neural Networks,” The Gradient, 2022. https://thegradient.pub/graph-neural-networks-beyond-message-passing-and-weisfeiler-lehman/. Accessed on: Mar. 02, 2023.