GRID ARCHITECTURE, PLANNING AND DESIGN JOURNAL
GRİD MİMARLIK, PLANLAMA VE TASARIM DERGİSİ

Vol. 8, No. 2, 2025, 621-639 / Cilt 8, Sayı 2, 2025, 621-639 / DOI: 10.37246/grid.1591809

A case study of a machine learning usage in design: conceptual models from graph-based GANs

Mustafa Koç[1], ORCID: 0000-0001-8131-8878
İmdat As[2], ORCID: 0000-0002-0692-4357

Abstract

This paper presents a novel machine learning (ML) pipeline that transforms architectural graph representations into fully rendered three-dimensional (3D) conceptual massing models. Unlike previous ML approaches that focus primarily on 2D floorplan generation, our method integrates multiple components into a single workflow: (1) graph-based input using HouseGAN++, (2) image-based shape extraction via custom MATLAB processing, (3) 3D model construction with FloorplanToBlender, and (4) diffusion model–based style transfer for visual enhancement. This end-to-end approach is distinctive in its combination of automated plan-to-volume conversion and aesthetic exploration through generative image synthesis. The results show that our pipeline enables efficient, multi-stage architectural ideation while significantly reducing manual effort. The proposed method contributes to early-stage design processes by accelerating concept development and offering stylistically diverse outputs from abstract spatial inputs.

Highlights

- New procedure for architects to create design alternatives from graph representations is presented.
- Conversion of node-graph representations of floor plans into 3D models is accomplished.
- Various styles are applied to rendered images of 3D models by diffusion models.

Keywords

Machine learning; Conceptual design; Graph-based learning; Massing models; GAN

Article Information

Received:
26.11.2024

Received in Revised Form:
12.06.2025

Accepted:
25.06.2025

Available Online:
28.10.2025

Article Category

Research Article

Contact

1. İstanbul Technical University, İstanbul, Türkiye

kocm21@itu.edu.tr

2. İstanbul Technical University, İstanbul, Türkiye

ias@itu.edu.tr

Koç, M. & As, İ. (2025). A case study of a machine learning usage in design: conceptual models from graph-based GANs. *GRID Architecture Planning and Design Journal*, 8(2), 621-639.

GRID ARCHITECTURE, PLANNING AND DESIGN JOURNAL
GRİD MİMARLIK, PLANLAMA VE TASARIM DERGİSİ

Vol. 8, No. 2, 2025, 621-639 / Cilt 8, Sayı 2, 2025, 621-639 / DOI: 10.37246/grid.1591809

Makine öğrenmesinin tasarım alanında kullanımı: grafik tabanlı GAN'lardan kütle modeli oluşturma

Mustafa Koç[1], ORCID: 0000-0001-8131-8878
İmdat As[2], ORCID: 0000-0002-0692-4357

Anahtar Sözcükler

Makine öğrenmesi; Konsept tasarım; Grafik-tabanlı derin öğrenme; Kütle modelleri; GAN

Öz

Bu makale, mimari grafik gösterimlerini üç boyutlu (3B) kavramsal kütle modellerine dönüştüren bir makine öğrenimi (ML) sistemi sunmaktadır. Öncelikle 2B kat planı oluşturmaya odaklanan önceki ML yaklaşımlarının aksine, yöntemimiz birden fazla bileşeni tek bir iş akışına entegre eder: (1) HouseGAN++ kullanarak grafik tabanlı girdi, (2) özel MATLAB görüntü işleme metotları ile görüntü tabanlı şekil çıkarma, (3) FloorplanToBlender ile 3B model oluşturma ve (4) görsel geliştirme için difüzyon model tabanlı stil transferi. Bu uçtan uca yaklaşım, otomatik plan-hacim dönüşümü ve üretken görüntü sentezi yoluyla estetik keşfin birleşiminde farklılık göstermektedir. Sonuçlar, geliştirilen sistemin manuel çabayı önemli ölçüde azaltırken verimli, çok aşamalı mimari fikir oluşturmayı sağladığını göstermektedir. Önerilen yöntem, kavram gelişimini hızlandırarak ve soyut mekânsal girdilerden stilistik olarak çeşitli çıktılar sunarak erken aşama tasarım süreçlerine katkıda bulunur.

Makale Bilgileri

Alındı:
26.11.2024

Revizyon Kabul Tarihi:
12.06.2025

Kabul Edildi:
25.06.2025

Erişilebilir:
28.10.2025

Makale Kategorisi

Araştırma Makalesi

Öne Çıkanlar

- Mimarların grafik gösterimlerinden tasarım alternatifleri oluşturmaları için yeni prosedür sunuldu.
- Kat planlarının düğüm-grafik gösterimlerinin 3B modellere dönüştürüldü.
- Difüzyon modelleri tarafından 3B modellerin son görsellerine çeşitli stil atamaları yapıldı.

İletişim

1. İstanbul Teknik Üniversitesi, İstanbul, Türkiye

kocm21@itu.edu.tr

2. İstanbul Teknik Üniversitesi, İstanbul, Türkiye

ias@itu.edu.tr

Koç, M. & As, İ. (2025). Makine öğrenmesinin tasarım alanında kullanımı: grafik tabanlı GAN'lardan kütle modeli oluşturma. *GRID Mimarlık Planlama ve Tasarım Dergisi*, 8(2), 621-639.

INTRODUCTION

Architecture is a complex and multifaceted field that demands the integration of spatial, aesthetic, and functional aspects. In recent years, technological advancements have significantly influenced architectural design, reshaping traditional approaches and introducing new ways for architects to conceptualize and create. The progression from isolated data sources to integrated, information-rich datasets has transformed design workflows. This evolution highlights a growing trend toward interdisciplinary collaboration and informed, data-driven design strategies. Architects now incorporate diverse information sources into their processes, including urban analysis, climate modeling, and 3D scanning, along with internally produced data from simulations, parametric modeling, and sensor technologies (Tamke, Nicholas, & Zwieryzycki, 2018). As a result, the field is experiencing a major shift with data becoming a pivotal element in architectural designs.

Many machine learning (ML)-based approaches in architecture are still limited to producing 2D floor layouts in spite of developments in the field. Although these methods provide useful answers for particular problems, they frequently fall short in addressing the three-dimensional complexity of architectural design. Machine learning, with its ability to uncover patterns and generate novel solutions, holds the potential to transform architectural workflows by addressing long-standing challenges and enabling scalable, efficient, and creative approaches (Chaillou, 2019). The incorporation of ML into 3D spatial design is still in its infancy and offers an opportunity to explore its application in creating conceptual massing models; a fundamental component of early-stage architectural design.

By automating the generation of 3D conceptual massing models using graph representations, this work presents a novel method for incorporating machine learning (ML) into the architectural design process. Graphs, which can encode relationships and spatial hierarchies, serve as a powerful abstraction for architectural designs.

A graph representation of an architectural building serves as the starting point for our approach. Using HouseGAN++, these graphs are utilized to create a variety of floor plan layouts that are functional and optimized. The generated floor plans are then transformed into volumetric 3D models, which let architects see structural shapes and spatial interactions. Finally, the conceptual massing models are refined by stacking and composing them to form cohesive design representations. This process represents an early attempt to automate the conceptual design phase, demonstrating how machine learning can bridge the gap between abstract data structures and tangible architectural outputs.

Our approach differs from existing work in three key ways: (1) it adapts and extends HouseGAN++ through custom pre- and post-processing to enable multi-level floorplan integration; (2) it introduces a semi-automated conversion of 2D plans into 3D massing via architectural image processing; and (3) it incorporates diffusion models to stylistically diversify conceptual outcomes. Compared to prior studies that stop at 2D plan generation or use GANs for stylistic experiments without architectural grounding, our pipeline offers a structured, end-to-end system capable of producing architecturally plausible and visually diverse conceptual designs. This work contributes to both the methodological expansion of generative design tools and the practical acceleration of early-phase architectural workflows.

By expanding the application of ML into the realm of conceptual massing, this study contributes to the ongoing digital transformation of architectural design protocols. The ability to translate graph-based representations into volumetric models not only accelerates the iterative design process but also provides a new lens through which architects can explore design possibilities. Furthermore, the integration of advanced ML techniques into architectural workflows offers a glimpse into a future where computational systems can play a creative role in shaping the built environment.

Background

Without explicit programming, computers can already recognize patterns in data and make well-informed decisions thanks to machine learning (ML), a subfield of artificial intelligence (Atalay & Çelik, 2017). With the exponential growth of digital data and computational power, ML algorithms have been increasingly adopted across various domains—including science, healthcare, engineering, and architecture (Wang, 2022; Topuz & Alp, 2023). In recent years, the creative sectors have used ML-based solutions. A review of machine learning usage in architecture by Topuz and Alp (2023) showed that ML techniques are used in various areas in architecture such as sustainability, historical and cultural structures, smart building design, space design, and digital fabrication (Topuz & Alp, 2023). Tools such as RunwayML allow designers to apply complex ML models without programming expertise, reducing barriers to entry (https://docubase.mit.edu/tools/runwayml/). Other examples for software tools using machine learning are "Magnetizing Floor Plan Generator," "Syntactic," and "Termite Nest." These tools allow algorithmic exploration of architectural layouts, but the majority are only able to produce 2D outputs and are not integrated into three-dimensional workflows (https://www.food4rhino.com/en).

Generative Adversarial Networks (GANs) which is introduced by Goodfellow et al. (2014) is one of the models that employs deep learning techniques and has developed significant popularity in the domain of unsupervised machine learning, particularly for the purpose of creating new data (Goodfellow et al., 2014). With the advent of GANs, it has become possible to output highly sophisticated human faces, urban scenes etc. (Nauata et al., 2021). The differentiating factor of GANs from other neural networks is the two conflicting neural networks known as GANs two sub networks: generator and discriminator. These two networks work together: the generator which generates new data and the discriminator which tells whether the data generated is apt or not. In computer vision, GANs have been instrumental in tasks such as generating realistic images, enhancing image resolution, and creating entirely new styles of visual content. Beyond visual tasks, GANs have also been used in natural language processing, music composition, and even creative design applications. In architecture and design, GANs have proven to be valuable tools for enhancing creativity and streamlining workflows. For instance, they have been used to generate urban layouts, simulate interior spaces, and conceptualize new architectural forms. In landscape architecture, GANs have shown potential by producing realistic designs for house gardens and backyards, enabling rapid exploration of design possibilities (Author et al., 2023). Their ability to work with complex and diverse datasets makes them particularly effective in design disciplines where attention to detail and contextual accuracy are critical. By bridging the gap between technical capability and artistic expression, GANs are helping designers reimagine what is possible in both form and function.

Diffusion model image generation, on the other hand, is a process that combines a neural network architecture with the capability of producing images. These models iteratively denoise random patterns into coherent images, operating within a latent space for increased efficiency and resolution. In order to create an image using a diffusion model, the first step is working with the network in relation to an image. This image normally consists of a minimal pattern that is to be termed a seed, which can be regarded as the minimal building block of the picture to be generated. The network then progressively edits the pixels of the wider image based on the patterns in the smaller one in a process that augments the patterns further across the larger surface. One of the main benefits of a diffusion modes is its ability to generate a large number of images that correlate with the patterns established in the seed. This is due to the fact that the network concept is able to 'upscale', i.e., to continuously improve the image produced through alteration of its resolution and pattern implicating a greater amount of detail. One example of a diffusion model image generation system is the DALL-E model, developed by OpenAI (OpenAI, 2021). This model, for instance, uses a 12-billion parameter version of the GPT-3 transformer network to generate high-resolution images from text descriptions. Overall, diffusion models are powerful tools for generating new, synthetic images based on a given seed pattern. In design applications, diffusion models offer nuanced control over visual styles and have demonstrated strong performance in architectural rendering, material visualization, and speculative aesthetics.

In this study, we build upon our previous work (Author et al., 2019), where we developed a function-driven deep learning approach to identify high-performing subgraphs from Building Information Models (BIM) of residential buildings. These subgraphs captured essential spatial and programmatic relationships by translating architectural layouts into graph representations, where nodes represented rooms and edges denoted functional connections. Inspired by graph mining approaches in fields such as chemistry, we identified frequently recurring substructures associated with desirable spatial qualities. Every new design was put through a validation procedure, which identifies viable solutions and filters out designs that aren't feasible. The deep neural network successfully identified latent sub-graphs with high-performance and integrated them into novel conceptual design compositions such as shown in the Fig.1. These subgraphs now serve as inputs to our current study, where we extend the approach into three dimensions and automate the transformation of these graphs into volumetric architectural massing models.

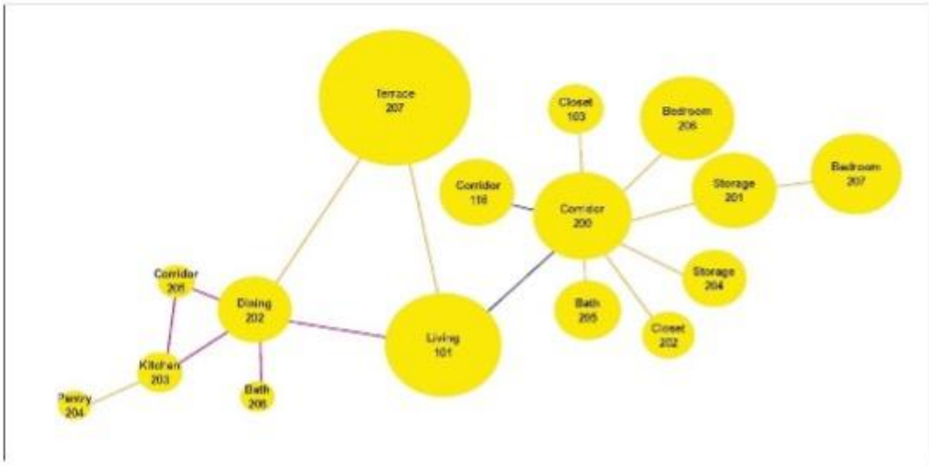

Figure 1: Node graph of high-performance subgraphs generated from two houses.

Our study aims to generate 3D massing models from these subgraphs. We examined existing machine learning technologies designed exclusively for the architectural design space. We transformed graphs into floorplans using HouseGAN++ (2020), and then used the Floorplan-to-Blender3D (2021) model to produce 3D massing models from the resulting floor plan layout images (Author et al., 2021). In this context, graph representations are spatial diagrams in which nodes represent programmatic spaces, such as bathrooms, and edges reflect adjacency or functional links, such as a door. These graphs differ from traditional architectural diagrams by serving as structured data for ML systems while providing information about room types, connections, and layout regulations (Author et al., 2021). This study primarily employs graphs as input data to develop three-dimensional design ideas rather than analyzing their functional performance.

However, the strategy for this study is to develop an automated process within a single workflow protocol. Algorithms HouseGAN++ and FloorplanToBlender utilized in this research involve manual intervention. The HouseGAN++ successfully generates floorplans, however they are all distinct floorplans for each floor level, and FloorplanToBlender generates 3D models of each floorplan independently after floorplan images are processed (Author et al., 2021). Selecting and stacking the appropriate floor levels into a single 3D massing model is done by hand, afterwards the level stacking is carried out automatically. Our objective is to develop a fully automated technique that can produce a multilayer 3D massing model in a single protocol by editing and improving the HouseGAN++ and FloorplanToBlender that we now utilize.

Methods and Tools

Bubble diagrams are utilized by architects to convert a particular building program into a floor plan, which they would subsequently turn into three-dimensional architectural represantations. In this study, we show how this age-old method, which involves transforming architectural graph data into three-dimensional conceptual massing models, might possibly be automated. We build on our previous machine learning work, in which we uncovered latent topological elements, or essential building blocks, in architectural building data, combined them into new designs, assessed their viability, and investigated GANs to produce unprecedented new designs (Author et al., 2021).

After converting a graph into a floorplan using HouseGAN++, MATLAB is used to extract shape features like doors, walls, and rooms from the image of floorplan and they are saved to a text file so they could be extruded into a 3D model. These floorplan images were then transformed into three-dimensional conceptual massing models using the Floorplan-to-Blender3D process.

A type of generative adversarial network (GAN) called HouseGAN++ is intended to produce a variety of floor plan layouts given a bubble diagram as input (Nauata et al., 2021). GANs represent a significant advancement in deep learning, offering potential for diverse applications, such as creating novel images. HouseGAN++ was developed using a dataset of 60,000 floorplans sourced from the RPlan database, which includes vector graphics floorplan diagrams (Nauata, et al., 2021). The software uses bubble diagrams as input sources, where a room with its type are respresented by a node and an edge represents a functional connection. Convolutional message passing neural networks (Conv-CPN) are used to convert the floorplans into graphs. Consequently, HouseGAN++ can produce a range of floorplans starting from simple input graphs. In this study, we use HouseGAN++ to convert existing graphs from our earlier research work (Author et al. 2019) into novel floorplans.

Conv-MPN utilizes a relational graph framework and is capable of interpreting graph representations that include data structured as nodes and edges. Conv-MPNs represent features with latent 3D volumes and use matrix multiplications with convolutions to deliver messages, as opposed to ordinary graph neural networks (GNNs), which numerically encode geometry information by 1D vectors (Author et al., 2021). Conv-MPN utilizes a graph that has a defined spatial embedding and represents every node as a feature volume. This volume is subsequently processed by a shared three-layer convolutional neural network (CNN) to generate a room segmentation mask, which is then given to a discriminator. The discriminator executes a series of tasks in the opposite sequence.

To support architectural needs, we introduced two modifications to the standard HouseGAN++ and FloorplanToBlender pipelines. First, we started with a pre-processing step that merges independently generated floorplans for each level into one cohesive set, allowing for semantic stacking of multilevel floorplans. Second, we used a MATLAB parser that remaps extracted shape features, e.g., windows, doors, from adjacency matrices, improving volumetric interpretation precision. While some of the pipeline remains semi-automatic, these additions reduce manual labor and set the stage for future automation.

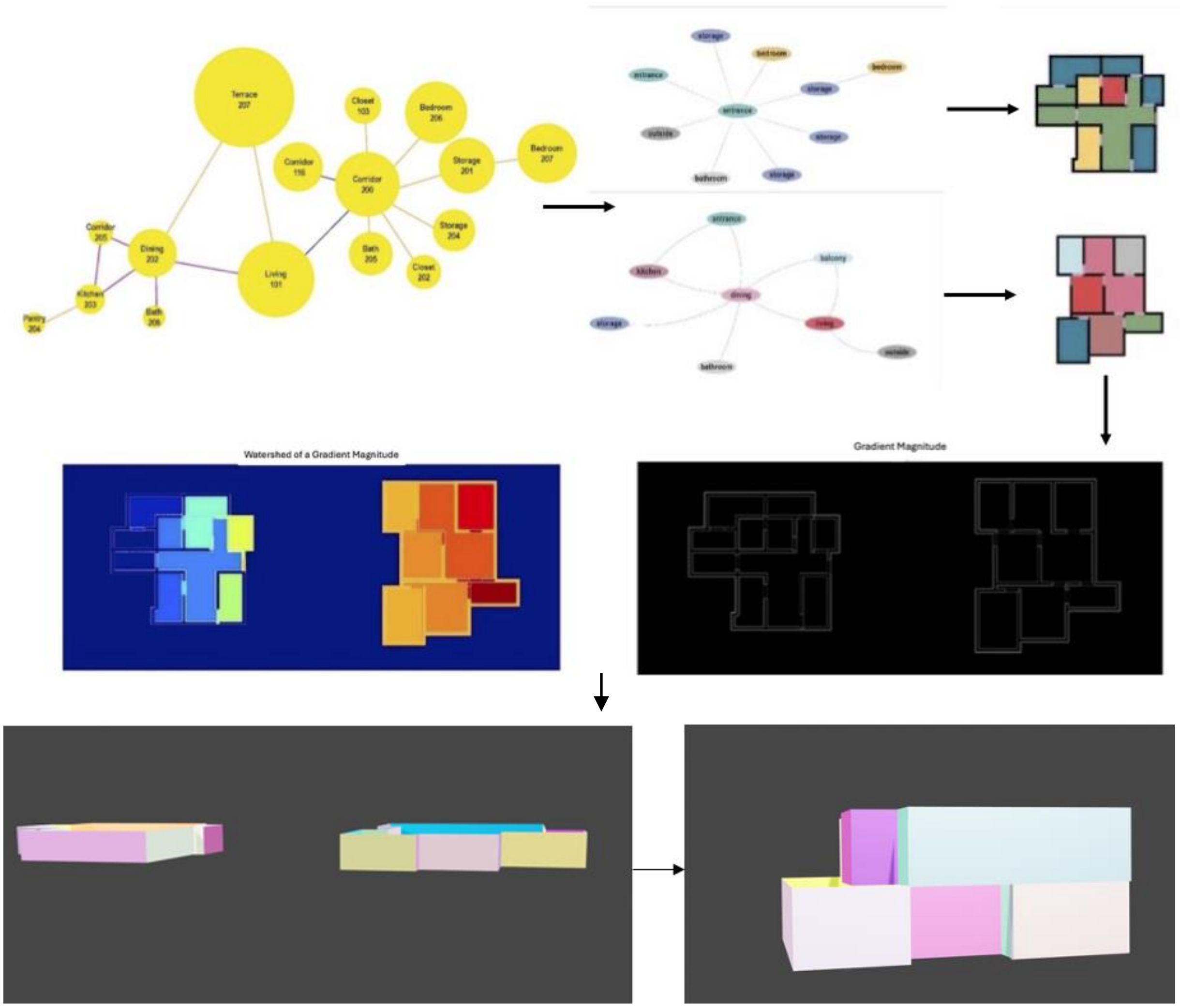

Figure 2: Flow diagram of the study.

In order to create a 3D model later, the output of House-GAN++ is imported into MATLAB to extract different shape features from the floorplan images. Rooms, walls, doors, and other features can be identified and extracted from images using distance transform, Harris corner detection, and watershed techniques (Grebtsew, 2021). The watershed algorithm separates objects in an image by treating pixel intensity as topographic elevation, flooding from marked regions until the boundaries between them meet along watershed lines. In theory, the grayscale image is interpreted as a topographic surface, with high intensity representing hills and low intensity representing valleys (OpenCV, 2021). The segmentation results are then produced by detecting the surrounding pixels of each local minima until they converge. Images are analyzed using the Harris corner identification technique to locate room and wall corners following watershed segmentation. A gaussian filter is applied to the grayscale image to smooth out the noises in each pixel. For computing the corner strength function, a 3x3 window is used. A feature descriptor is computed for pixels that are local maxima within a certain window and over the threshold (Tyagi, 2019). The grayscale image is processed to calculate the Euclidean distance transform after the Harris corner detection is applied. The distance transform gives each pixel in the image a number that represents the separation between that pixel and the image's closest nonzero pixel. The transform's output displays the distance between each point and the nearest border. After collecting shape features from floorplan images, Grebtsew's model uses these elements to generate 3D massing models of floorplans (Grebtsew, 2021). After creating the 3D massing model of the generated floorplans, diffusion models, e.g., FreewayML (2022) and DreamStudio (2022), are used for exploring various architectural styles on our 3D massing models.

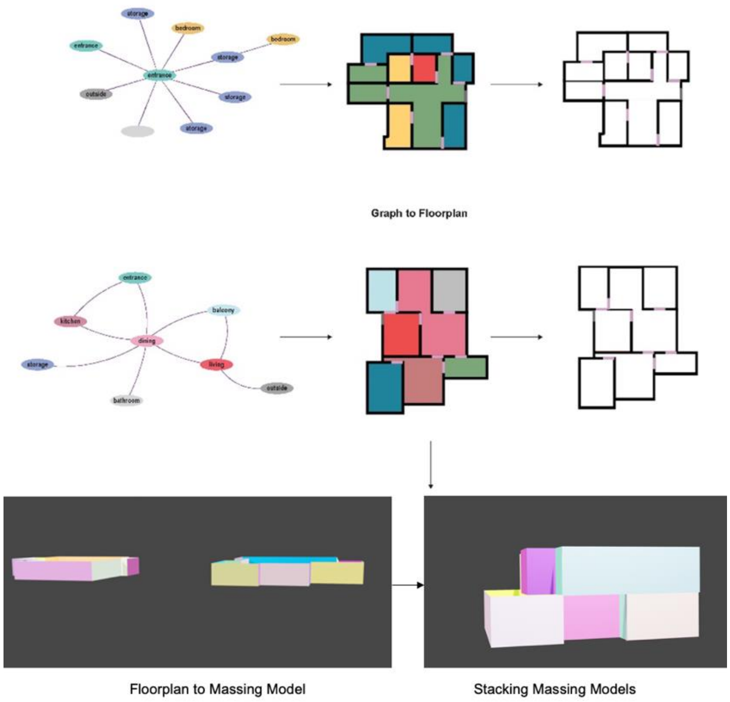

Figure 3: Overall Schema of the Study

Case Study

Some of the high-performing subgraphs are converted from the previously mentioned work (Author et al., 2018) into 3D conceptual massing models for this case study. We then used different architectural styles on the massing models. Specifically, Fig. 1 displays the graphs we began. The workflow is as follows (Fig.2):

a) Selecting a subgraph with high performance
b) Use subgraph as input to HouseGAN++
c) Extrude floorplans and stack them to create a 3D model
d) Apply a style to the 3D model with AI diffusion models

a. Selecting a subgraph with high performance

The subgraphs in Fig.4 and Fig.5 which were part of the larger graph of a house (fig.1) from our earlier study are chosen for the study (Author et. al., 2018). We chose this particular group of subgraphs because it reflected a wide variety of the original building program.

Vol. 8, No. 2, 2025, 621-639 / Cilt 8, Sayı 2, 2025, 621-639 / DOI: 10.37246/grid.1591809

b. Use subgraph as input to HouseGAN++

To convert the graph into a floorplan, we used the subgraph as input data for the House-GAN++ software tool (Nauata et al., 2021) where room type is encoded by nodes in the networks, whereas spatial adjacency is encoded by edges. The algorithm uses Conv-MPN, which replaces the traditional 1D latent vector with a 3D feature volume and uses convolutional layers instead of fully connected layers to encode messages between nodes (Zhang, Nauata, & Furukawa, 2020). The algorithm's architecture is predicated on the combination of a conditional GAN with a relational GAN which provides a framework for improving the preferred metric by meta-optimizing the refinement scheme during testing (Nauata et. al., 2021). Three key components of the HouseGAN++ software's architecture are: 1) Conv-MPN feature pooling is reformulated to allow nodes and edges to exchange features; 2) A 2D segmentation mask is accepted by each node or edge as an additional input constraint with a new loss associated with it; and 3) Edges carry features for door construction in addition to nodes (Nauata et al., 2021). They represented each element (room or door) using a 12-dimensional one-hot encoded vector, where each dimension corresponds to one of 10 room types or 2 door types. Only one entry in this encoding is set to 1 (signaling the type); all other entries are set to 0. As an input condition for every door or room, the model used ground-truth conditional training, which involves randomly choosing a ground-truth layout, initializing a relational graph, and setting a ground-truth segmentation mask. By passing previously created layouts as input conditions, they have a 50% chance of improving design (Nauata et al., 2021). A noise vector and a room type are used to initialize each node in House-GAN++, and these are converted into an $8 \times 8 \times 16$ feature volume. For every node or edge, the relational generator receives an extra $64 \times 64 \times 2$ condition image. The generator learns to maintain the segmentation mask, which is provided by the first channel. If the segmentation mask is supplied, the second channel becomes 1 for each pixel; if not, it becomes 0. The condition image is converted to $8 \times 8 \times 16$ using a 3-layer CNN, and then concatenated to the original feature.

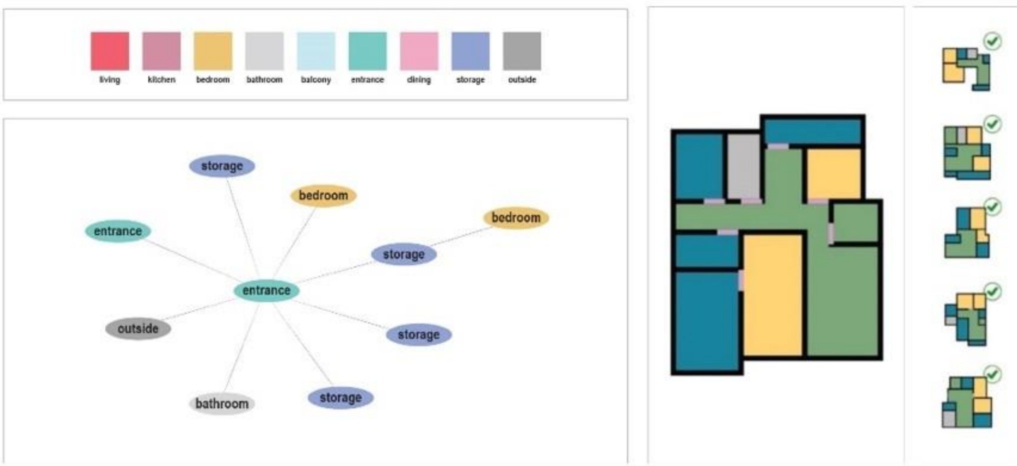

Figure 4: Generated Floorplans from Node Graph Example 1.

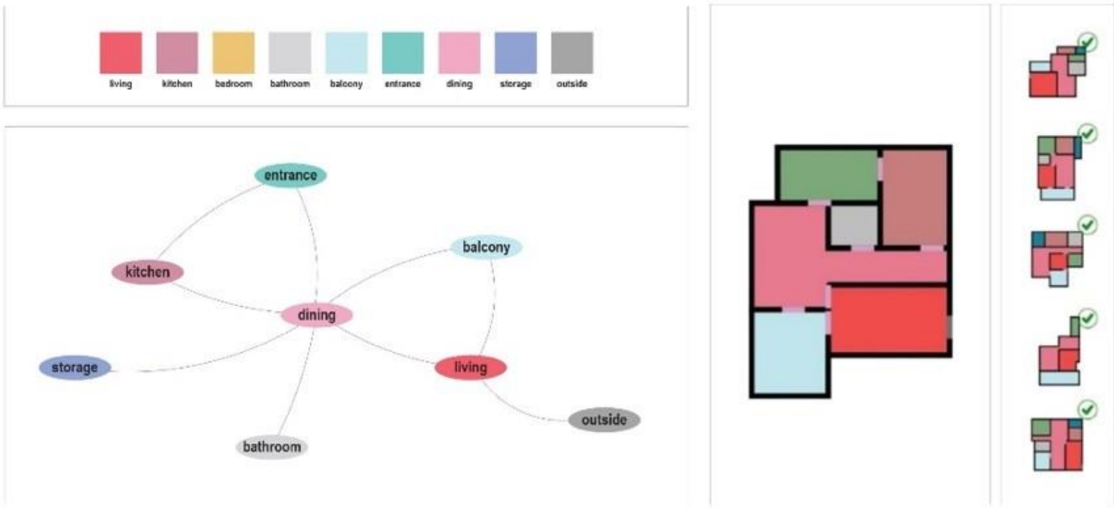

Figure 5: Generated Floorplans from Node Graph Example 2.

During this procedure, the floorplan with high function produced by HouseGAN++ is selected and used to create a three-dimensional massing model in the following stage.

c. Extrude floorplans into 3D

A 3D massing model is generated by identifying walls, doors, windows, and rooms in the generated floorplan images. We chose the image-to-3D model approach from Github and used MATLAB to make the floorplan images machine-readable (Grebtsew, 2021). In order to generate a text file that contains shape data such pixel coordinates, floorplans are analyzed for form and room detection (Author et al., 2021). The text files are then utilized to generate a 3D massing model in Blender3D. Using the watershed and Harris corner identification algorithms, as well as the distance transform, flooplans are analyzed to detect rooms and walls based on their shapes.

Considering a picture as a topological landform in geodesy is the fundamental principle behind the watershed segmentation approach, which is based on the mathematical morphology of topological theory (Wen et. al., 2022). In the watershed algorithm, each local minimum in the image forms a catchment basin along with its surrounding region. The gray value of each pixel is treated as elevation, and the watershed lines mark the boundaries where different basins meet. For applying the watershed segmentation, we firstly convert the images to grayscale. Next, we mark the foreground objects using gradient magnitude as the segmentation function, then we calculate the background markers, and lastly, we calculate the segmentation function's watershed transform (Author et al., 2021). The watershed algorithm is used because it is effective at segmenting objects that are touching or closely positioned.

Harris and Stephens (1988) essentially determine the difference in intensity for displacement of (u, v) in all directions in Harris corner detection, which is represented as follows:

$$E(u.v) = \Sigma_{x,y} w(x,y)[I(x+u, y+v) - I[x,y]]^2$$

The window function (w), which can be either a Gaussian or rectangular window, assigns weights to the pixels underneath. For corner detections, the E(u, v) function must be maximized. To accomplish this, the second term must be maximized which is accomplished by applying Taylor Expansion to the given equation (Harris & Stephens, 1988). The image derivatives in the x and y axes were then obtained to generate a score that establishes whether or not a window has a corner.

After shapes are determined in images, the features are recorded to a text file to construct a 3D model. Grebtsew's approach employs shape features to generate 3D massing models in Blender3D. The model employs a server using the Swagger API to convert images to 3D using the author's floor to blender library (FTBL) (Grebtsew, 2021).

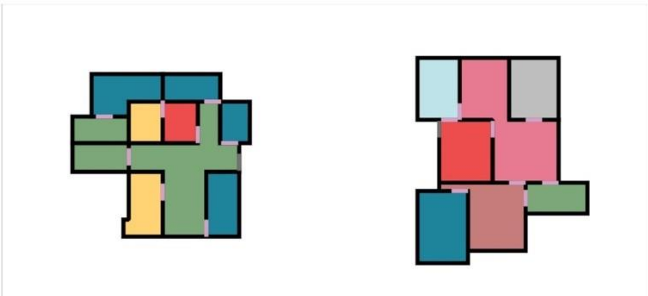

Figure 6: Floorplans of Different Floor Levels of Same Building.

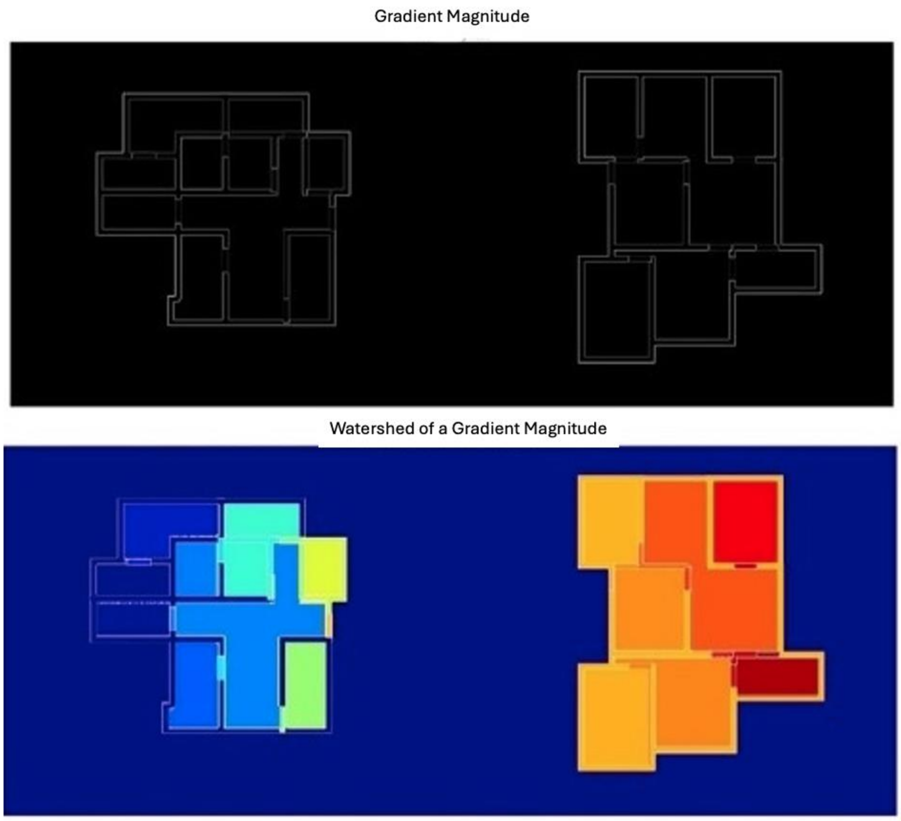

Figure 7: Image Processing for Detecting Shape Features and Transforming Floorplans to Generate a 3D Model.

After extracting the properties of the floors, we passed them to the FloorplanToBlender algorithm to generate the 3D model. The 3D model is created using the processed floorplan image and the spatial features extracted during image analysis.

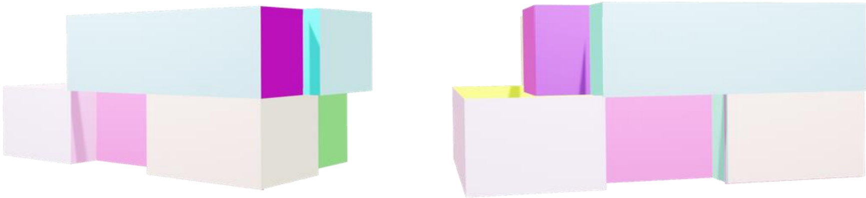

Figure 8: Stacked 3D Massing Models as Final Conceptual Model.

We adjusted the algorithm for extruding 3D models from floorplans in order to extrude and stack the floorplans automatically. The floors are combined in such a way that the first wall of each floor starts from the origin of the coordinate system. The stacking conditions can be adjusted according to any determined condition, such as elevators, staircases, etc., required by the design.

d. Style transfer with diffusion models

After the 3D models were created, we applied various architectural styles to the model viewed from different angles. We tested few style transfer algorithms. At first, we explored a style transfer algorithm developed by Gatsy, Ecker, and Bethge (2015), yet the results were not satisfactory. Then we tested various diffusion models. In particular, two of them yielded successful results: DreamStudio (2021) and FreewayML (2022). DreamStudio and FreewayML are built upon stable diffusion which is called a latent diffusion model. Unlike traditional diffusion models that operate in pixel space, latent diffusion models use optimization to reconstruct high-quality images from latent representaitons (Rombach et. al., 2022). In latent diffusion models, an autoencoder is trained to compress the data into lower-dimensional space which makes the process more efficient yet still preserves important visual details (Rombach et. al., 2022). Excessive spatial compression is not required because they are trained in latent space, which has superior scaling properties in terms of spatial dimensionality. One key advantage of latent diffusion models is that they can generate high-quality images efficiently with a single network pass through the compressed latent space. We tested different architectural styles from historical periods. Our results are illustrated in Figures 9 and 10. In Figure 11 we applied another style to an already style-transferred image.

Figure 9: Style Transfer with Dream Studio.

Figure 10: Style Transfer with FreewayML.

Figure 11: Style Transfer on the Diffusion Generated Image.

CONCLUSION

Using graph representations, this research aims to demonstrate a novel process for architects to generate design altenatives. Building on the earlier research of Author et al. (2018), we suggested an automated procedure that uses a series of computational processes to transform graphs into 3D conceptual massing models. Using the Watershed segmentation and Harris corner recognition methods, we first converted the graph into a floor plan, whose shape features—such as walls, doors, etc.—are automatically identified. The resulting floor plans were transformed into 3D models using FloorplanToBlender, and afterwards diffusion models were applied to assign varying architectural styles, enabling the exploration of a broad spectrum of design possibilities. Together, these components enable the rapid and semi-automated translation of abstract spatial data into meaningful, design-ready 3D outputs. In contrast to isolated ML applications, our pipeline emphasizes interoperability, from spatial logic to visual presentation, offering a flexible foundation for future tools in conceptual design.

To position our work within the broader context of ML-driven architectural design, we benchmark it against representative prior studies in terms of functionality, output dimensionality, and workflow automation. HouseGAN++ (Nauata et al., 2021) serves as a relevant baseline. It generates floorplans from graph-based inputs using a relational GAN, but outputs are limited to 2D plan layouts. While effective for programmatic layout synthesis, HouseGAN++ does not support volumetric translation, multi-floor integration, or stylistic rendering. In contrast, our system extends HouseGAN++ with additional modules that enable end-to-end processing: floorplan image analysis, 3D massing generation, and visual style transfer via diffusion models. Similarly, studies like Magnetizing Floor Plan Generator and Syntactic focus on interactive 2D floorplan creation based on user constraints or procedural rules (Egor, Sven, Martin, & Reinhard, 2019) (pirouzaan, 2018). These tools provide control over plan composition but do not scale to automated 3D modeling or aesthetic interpretation. While existing studies have contributed significantly to specific stages of architectural design automation, our work is unique in providing a unified, cross-domain pipeline from graph input to stylized 3D massing. This end-to-end integration is critical for supporting early-stage design iteration and for moving beyond static 2D layouts toward volumetric and visual experimentation.

Nevertheless, the protocol we describe in this study has certain drawbacks. First of all, it is limited to single-story structures. As a result, after 3D models are produced, structures with several storeys are automatically divided and stacked. Second, only box-shaped floorplans can be generated. In subsequent iterations of the study, we think we can overcome these restrictions. Despite these limitations, we think this study has considerable implications for architectural design. The system gives architects a tool to investigate a wide range of design alternatives early in the design process by automatically translating graphs into conceptual massing models. Furthermore, even though the technology creates conceptual models, designers can use them as a flexible starting point to develop more detailed architectural solutions. The integration of diffusion models adds an additional layer of creative freedom, allowing architects to envision the potential of their designs in varying stylistic contexts.

This study demonstrates the feasibility and transformative potential of automating the conversion of graph-based representations into conceptual 3D models. By integrating machine learning with architectural design, we open new avenues for creativity and efficiency, enabling architects to focus their expertise on developing and refining designs. As we look toward future iterations, we envision this protocol as a steppingstone toward a more dynamic and adaptive architectural design process. The ability to automate the conversion of graphs into conceptual massing models could provide advantages in the early phases of the design process. The designer can further enhance the ML system's fundamental conceptual massing models to produce sophisticated final designs (Author et al., 2021). With the fast evolution of generative models, particularly diffusion and transformer-based approaches, we acknowledge the short-term validity of the software and tools used in this study. However, the protocol we present in this study remains open to newer models and demonstrates a workflow where data-driven methods can enhance human intuition rather than replace it. Our goal is not to commit to a particular toolchain but to open a doorway to adaptive, ML-integrated conceptual design generation.

Conflict of Interest Statement | Çıkar Çatışması Beyanı

Araştırmanın yürütülmesi ve/veya makalenin hazırlanması hususunda herhangi bir çıkar çatışması bulunmamaktadır.

There is no conflict of interest for conducting the research and/or for the preparation of the article.

Financial Statement | Finansman Beyanı

Bu araştırmanın yürütülmesi ve/veya makalenin hazırlanması için herhangi bir mali destek alınmamıştır

No financial support has been received for conducting the research and/or for the preparation of the article.

Ethical Statement | Etik Beyanı

Araştırma etik standartlara uygun olarak yapılmıştır.

All procedures followed were in accordance with the ethical standards.

Copyright Statement for Intellectual and Artistic Works | Fikir ve Sanat Eserleri Hakkında Telif Hakkı Beyanı

Makalede kullanılan fikir ve sanat eserleri (şekil, fotoğraf, grafik vb.) için telif hakları düzenlemelerine uyulmuştur.

In the article, copyright regulations have been complied with for intellectual and artistic works (figures, photographs, graphics, etc.).

Author Contribution Statement | Yazar Katkı Beyanı

AUTHOR 1: (a) Idea, Study Design, (b) Methodology, (c) Literature Review, (h) Writing Text.

AUTHOR 2: (a) Idea, Study Design, (b) Methodology, (d) Supervision, (h) Writing Text, (i) Review

REFERENCES

As, İ., Pal, S., & Basu, P. (2018). Artificial Intelligence in Architecture: Generating Conceptual Design via Deep Learning. International Journal of Architectural Computing, 16(4), 306-327.

As, I., Pal, S., & Basu, P. (2019). Composing frankensteins: Data-driven design assemblies through graph-based deep neural networks. In the 107th Annual Meeting BLACK BOX: Articulating Architecture's Core in the Post-Digital Era. Pittsburgh, PA, USA, ACSA

Atalay, M., Çelik, E. (2017). Artificial Intelligence and Machine Learning Applications in Big Data Analysis. Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi 9 (22) 155-172.

Chaillou, S. (2019). AI + Architecture: Towards a New Approach. Harvard University, Graduate School of Design. Boston: Harvard University.

Egor, G., Sven, S., Martin, D., & Reinhard, K. (2019). Computer-aided approach to public buildings floor plan generation.Magnetizing Floor Plan Generator. 1st International Conference on Optimization-Driven Architectural Design (pp. 132-139). Amman: Elsevier Procedia.

FreewayML. (2022). freewayML. Retrieved from freewayML: https://www.freewayml.com/about-us stability.ai. (2021). DreamStudio. Retrieved from DreamStudio: https://dreamstudio.ai

Gatsy, L. A., Ecker, A. S., & Bethge, M. (2015, September 2). arxiv. Retrieved from Cornell University: https://arxiv.org/abs/1508.06576

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y. (2014, June 10). Generative Adversarial Networks. Retrieved from arxiv: https://arxiv.org/abs/1406.2661

Grebtsew. (2021, May 7). FloorplanToBlender3d. Retrieved from Github: https://github.com/grebtsew/FloorplanToBlender3d

Harris, C. G., Stephens, M. J. (1988). A Combined Corner and Edge Detector. Alvey Vision Conference.

Karabağlı, K., Koç, M., Basu, P., As, İ. (2021). A Machine Learning Approach to Translate Graph Representations into Conceptual Massing Models. Sigradi 2021: Designing Possibilities, Ubiquitous Conference (pp. 191-202). Colombia: Perkins & Will

Nauata, N., Hosseini, S., Chang, K.-H., Chu, H., Cheng, C.-Y., & Furukawa, Y. (2021, March 3). House-GAN++: Generative Adversarial Layout Refinement Networks. Retrieved from Cornell University: https://arxiv.org/abs/2103.02574

OpenAI. (2021, January 5). DALL·E: Creating Images from Text. Retrieved from OpenAI: https://openai.com/blog/dall-e/

OpenCV. (2021, April 2). Image Segmentation with Watershed Algorithm. Retrieved from Open-Source Computer Vision: https://docs.opencv.org/4.5.2/d3/db4/tutorial_py_watershed.html

Pirouzaan. (2018, October 10). SYNTACTIC. Retrieved from food4rhino: https://www.food4rhino.com/en/app/syntactic

Rombach, R., Blattman, A., Lorenz, D., Esser, P., Ommer, B. (2022). High-Resolution Image Synthesis With Latent Diffusion Models. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Tamke, M., Nicholas, P., & Zwieryzycki, M. (2018). Machine Learning for architectural design: Practices and infrastructure. International Journal of Architectural Computing.

Topuz, B., Alp, N.Ç. (2023). Machine Learning in Architecture. Automation in Construction.

Tyagi, D. (2019, March 16). Introduction to Harris Corner Detector. Retrieved from Data Breach: https://medium.com/data-breach/introduction-to-harris-corner-detector-32a88850b3f6

Wang, Q. (2022). Machine Learning in Architecture. Lund: Lund University.

Wen, T., Tong, B., Liu, Y., Pan, T., Du, Y., Chen, Y., Zhang, S., (2022). Review of research on the instance segmentation of cell images. Computer Methods and Programs in Biomedicine.

Zhang, F., Nauata, N., & Furukawa, Y. (2020, March 31). Conv-MPN: Convolutional Message Passing Neural Network for Structured Outdoor Architecture Reconstruction. Cornell University: https://arxiv.org/abs/1912.01756.

BIOGRAPHIES OF THE AUTHORS

Mustafa KOÇ

Mustafa Koç is an engineer who works in inter/trans disciplinary studies. After graduated from Bilkent University, completed his master's degree in both engineering and architecture. During his maser degrees, he focused on machine learning and its applications in various disciplines. Currently he is a Ph.D. candidate at İstanbul Technical University in Architectural Design Computing. He is studying machine learning, synthetic data, and decision support systems.

İmdat AS

Imdat As is an architect, entrepreneur, academician, and the recipient of the prestigious Turkish Scientific Research Council (TUBITAK) 2232 Grant and Fellowship. He leads the City Development through Design Intelligence (CIDDI) lab at Istanbul Technical University (ITU). His work focuses on new technologies that shape the morphology of the future city. He received his doctoral degree from the Harvard University Graduate School of Design (GSD), his M.Sc. in architecture from the Massachusetts Institute of Technology (MIT), and his B.Arch. from the Middle East Technical University (METU) in Ankara, Turkey. He co-authored, Dynamic Digital Representations in Architecture: Visions in Motion (Taylor & Francis, 2008), and two books on AI in Architecture (Taylor & Francis, 2021 and Elsevier, 2022). He founded Arcbazar.com in 2011 -- a crowdsourcing platform for architectural design. Arcbazar has been featured as one of the "Top 100 Most Brilliant Companies" by Entrepreneur Magazine.