VIVA-Tech International Journal for Research and Innovation ISSN(Online): 2581-7280

Volume 1, Issue 9 (2026)
PP 01-08

VIVA-TECH IJRI
ANNUAL RESEARCH JOURNAL
VIVA-TECH INTERNATIONAL JOURNAL FOR RESEARCH AND INNOVATION
ANNUAL RESEARCH JOURNAL
ISSN(ONLINE): 2581-7280

PlanVista3D

Om Koli[1], Hamza Khan[2], Harsh Jethwa[3], Prof Akshata Raut[4]

[1]*(Department of Computer Engineering, VIVA Institute of Technology, India)*
[2]*(Department of Computer Engineering, VIVA Institute of Technology, India)*
[3]*(Department of Computer Engineering, VIVA Institute of Technology, India)*
[4]*(Assistant Professor, Department of Computer Engineering, VIVA Institute of Technology, India)*

Abstract : *This paper presents a rule-based approach for automatic generation of three-dimensional (3D) multi-floor building models from architectural blueprint data. The proposed system takes standard CAD/BIM inputs such as DXF files and converts them into structured 3D building geometry without relying on machine learning models. The methodology focuses on extracting architectural elements including walls, floors, and storey information using geometric and topological rules. Each floor plan is processed independently and then vertically aligned based on detected elevation data to construct a complete multi-floor 3D model. The system is implemented using Python-based processing and integrated with 3D modeling software to generate accurate and editable 3D outputs. By eliminating the dependency on large training datasets, the proposed approach ensures transparency, consistency, and ease of adaptation for different building layouts. Experimental results demonstrate that the method can successfully generate realistic multi-floor 3D building models from real-world architectural drawings. This work provides a practical and efficient solution for automated building modeling, with potential applications in architectural visualization, digital construction workflows, and smart city planning.*

Keywords - *Architectural Blueprint, Automated 3D Modeling, DXF, Multi-floor Building, Rule-based System.*

I. INTRODUCTION

The rapid growth of digital construction and architectural visualization has increased the demand for automatic and accurate conversion of 2D architectural drawings into three-dimensional (3D) building models [4], [15]. Traditional 3D modeling workflows are largely manual, time-consuming, and require skilled expertise, which limits scalability for large or multi-floor building designs [4]. Most architectural designs are still exchanged in the form of 2D CAD drawings, particularly DXF files, making automated interpretation of such data an important research problem [14].

Several existing approaches attempt to address this problem using Building Information Modeling (BIM) tools or machine learning-based techniques [3], [6]. However, BIM-based solutions often require proprietary software and structured inputs, while machine learning approaches depend heavily on large labeled datasets and lack transparency in geometric decision-making. These limitations reduce flexibility and practical usability, especially in academic and low-resource environments [15].

This paper proposes a rule-based system that uses DXF files as the only input to generate 3D building models. The approach first focuses on generating an accurate 3D model of a single floor plan by extracting and interpreting geometric entities such as walls and enclosed regions. Subsequently, multiple single-floor models are integrated vertically to construct a multi-floor 3D building model using predefined elevation and alignment rules.The main contribution of this work lies in providing a transparent, dataset-independent, and fully automated pipeline for converting 2D DXF floor plans into complete multi-floor 3D building models.

II. REVIEW OF LITERATURE

Recent research demonstrates significant advancements in automated 2D-to-3D reconstruction systems, driven by the integration of computer vision, deep learning, and procedural modeling techniques. Existing studies explore a wide range of approaches, including depth estimation, semantic segmentation, diffusion models, and hybrid rule-based pipelines to convert architectural blueprints and floor plans into three-dimensional models. Techniques such as CNN-based feature extraction, UNet variants, diffusion-driven RGB-D synthesis, and photogrammetry-based reconstruction have shown promising results in generating accurate 3D representations from 2D inputs. However, many methods rely heavily on large labeled or synthetic datasets, require extensive preprocessing, or are limited to partial 2.5D reconstructions. Additionally, several approaches depend on cloud-based processing or paired data, restricting their applicability in secure or offline environments.

Unwei Zhou et al. [1] The proposed system leverages a combination of advanced computer vision and AI-based techniques to achieve realistic image processing and generation. Depth estimation is performed using MiDaS integrated with OpenCV, enabling the model to capture geometric information from images. Image decomposition is achieved through a CNN Autoencoder, which separates scene elements into meaningful layers for further processing. For text-driven image generation, the system utilizes the Stability AI API, which allows for reference-based image synthesis guided by textual prompts. Additionally, SAM segmentation is employed to isolate objects and regions of interest, while a Large Multi-view Gaussian Model facilitates multi-perspective understanding of the scene.

Saranya S et al. [2] In this work, several advanced technologies and methodologies were employed to analyze and model complex 3D structures. Computer-Aided Design (CAD) tools were utilized for geometric modeling, while Deep Learning and Computer Vision techniques enabled automated feature extraction and interpretation of visual data. For image-based analysis, frameworks such as Blender and OpenCV were employed alongside Convolutional Neural Networks (CNNs) for robust feature extraction. Traditional image processing algorithms, including Canny edge detection and the Hough Transform, were integrated with deep learning models to create a hybrid approach that leverages both data-driven and rule- based insights.

Hieu Trung Nguyen et al. [3] This study leverages state-of-the-art Diffusion Models and Computer Vision techniques to achieve RGB-D novel view synthesis for 3D scene reconstruction. The approach integrates a modified UNet architecture along with Depth-enhanced Camera Positional Encoding (DeCaPE) to effectively incorporate depth information into the model. The model demonstrates a strong dependency on synthetic datasets, which may not fully capture the variability of real-world scenarios. Moreover, it faces challenges in handling noisy floorplans, limiting its robustness and generalization when applied to practical, unstructured environments.

Sofia Feist et al. [4] This research explores the integration of CAD and Building Information Modeling (BIM) frameworks with advanced raster and vector processing techniques to facilitate automated 3D modeling and analysis. The methodology incorporates a range of deep learning and computational algorithms, including Convolutional Neural Networks (CNNs), Region-based CNNs (R-CNNs), and Generative Adversarial Networks (GANs), alongside graph-based modeling approaches. Geometric processing techniques such as extrusion and triangulation are employed to construct accurate 3D representations from 2D inputs.

Anna Wilson et al. [5] This study utilizes a combination of Artificial Intelligence (AI), Deep Learning, and Image Processing techniques to enable advanced 3D reconstruction and scene understanding. The approach leverages Convolutional Neural Networks (CNNs), Generative Adversarial Networks (GANs), and VGG19 for feature extraction and image- based modeling, while geometric and spatial consistency is ensured using RANSAC and Structure- from-Motion (SfM) techniques. Additionally, voxel-based approaches are employed to represent volumetric data for accurate 3D modeling. Despite these technological advancements, the methodology faces notable challenges, including high training data requirements and significant computational complexity, particularly when scaling to large datasets or high-resolution models. These gaps highlight the need for optimization and more efficient model architectures to enable practical deployment in real-world applications.

Lukas Kratochvila et al. [6] This work employs Deep Learning architectures such as MDA-UNet, MACU-Net, and U-Net3+ with asymmetric convolutions and attention mechanisms for pixel-wise semantic segmentation. Techniques like multi-task loss (focal + affinity + heatmap regression), post-processing, vectorization, and 2.5D reconstruction are used to generate accurate floorplan representations. However, the approach requires raster preprocessing, is limited to 2.5D accuracy, is sensitive to noisy or diverse floorplans, and shows strong dependence on datasets, highlighting areas for future improvement.

A.C. Barreiro et al. [7] This study utilizes deep learning for image-feature extraction, combined with CityJSON/CityGML integration, GIS data fusion, and classical reconstruction heuristics to extract geometry from floorplan images and align it with city-scale datasets like 3D BAG. The reconstructed building volumes are validated using IoU metrics. Despite its effectiveness, the approach requires representative training data and shows limitations in handling highly detailed interiors, indicating the need for more diverse datasets and improved reconstruction methods.

Vikram Voleti et al. [8] This work employs deep learning (3D CNNs) alongside Statistical Shape Models and particle distribution models for 3D shape regression, with PCA used for low-dimensional shape representation. Model-based data augmentation techniques such as kernel density estimation (KDE) and thin-plate spline (TPS) warping enhance the training dataset. The approach, however, requires pre-segmented data and is heavily dependent on large training datasets, highlighting challenges in data preparation and scalability.

Victoria M Baretto et al. [9] This study integrates OpenCV, Pillow, and NumPy- STL for 2D-to-3D conversion, combined with Unity3D and Vuforia SDK for AR-based real-time dynamic 3D models, deployed via Python and AndroidNativePlugin for mobile compatibility. The system effectively combines 2D-to-3D reconstruction with AR visualization. However, it suffers from image processing errors, inefficient image cleaning, and experiences a slight delay in AR display, indicating areas for optimization.

ORCID,Beatriz et al. [10] This work leverages photogrammetry and Structure from Motion (SfM) techniques, integrated with BIM and open-source tools like Regard3D, MeshLab, and Blender. Methods include SfM triangulation, point cloud densification, mesh reconstruction, and retopology for optimization. Despite these capabilities, the approach shows limited accuracy compared to laser scanning and faces scalability and file size challenges when applied to large heritage structures.

Fabrizio Ivan et al. [11] This study utilizes photogrammetry with smartphone cameras and illuminator kits, combined with color correction and denoising for image preprocessing. Unity3D is used for mesh processing, while Structure-from-Motion (SfM) and Multi-View Stereo (MVS) enable 3D reconstruction of small museum artifacts. The approach is effective for detailed modeling but requires a controlled rotation setup, highlighting a limitation in flexibility and ease of capture.

Monique Arkesteijn et al. [12] This study processes floorplan images and photos using CNNs and Graph Neural Networks (GNNs), combined with mesh processing and texture synthesis pipelines. The approach lifts floorplans to 3D meshes and propagates textures to unseen surfaces using GNN-based inference. However, it relies on paired floorplans and photos and shows limited generalization to complex or large commercial buildings.

João Poças Martins et al. [13] This work employs a hybrid approach combining image processing, deep learning, and rule-based heuristics in an ensemble pipeline, with vectorization and STL export for 3D modeling. Floorplan raster preprocessing, CNNs, and heuristic rules are used to assemble geometry into 3D vector models. The method is heavily dataset-dependent and requires large, labeled datasets for robust machine learning-based element detection.

Xiaobao Wei1 et al. [14] This study utilizes Computer-Aided Design (CAD) and vector processing, employing algorithms such as the Shape-Opening Graph (SOG), Priority Principle Algorithm, Edge Breaking Algorithm. The approach effectively processes vector floorplans but is limited to vector inputs, struggles with highly complex or irregular architectural styles, and often requires manual preprocessing

Ming Lu et al. [15] This work integrates RGB, LiDAR, and Kinect sensors with SLAM, Multi-View Stereo, SfM, and Deep Learning (CNNs), adhering to BIM standards like IFC and CityGML. Techniques include geometry, semantic, and topological modeling using grid and subdivision methods. The approach relies on the Manhattan assumption and is affected by occlusion, clutter, weak textures, and sensor noise. Additionally, there is a lack of benchmarks for topology, and indoor–outdoor integration remains limited.

III. METHODOLOGY

The proposed PlanVista3D system aims to automate the conversion of 2D architectural blueprints into fully navigable 3D walkthrough models using a combination of image processing, procedural modeling, and 3D rendering techniques. The system is designed to minimize manual modeling effort and reduce turnaround time by automatically interpreting spatial layouts from floor plans and generating accurate three-dimensional representations. By integrating blueprint input handling, parameter-driven 3D generation, model storage, and interactive visualization, the project provides a complete end-to-end solution for efficient architectural and operational visualization.

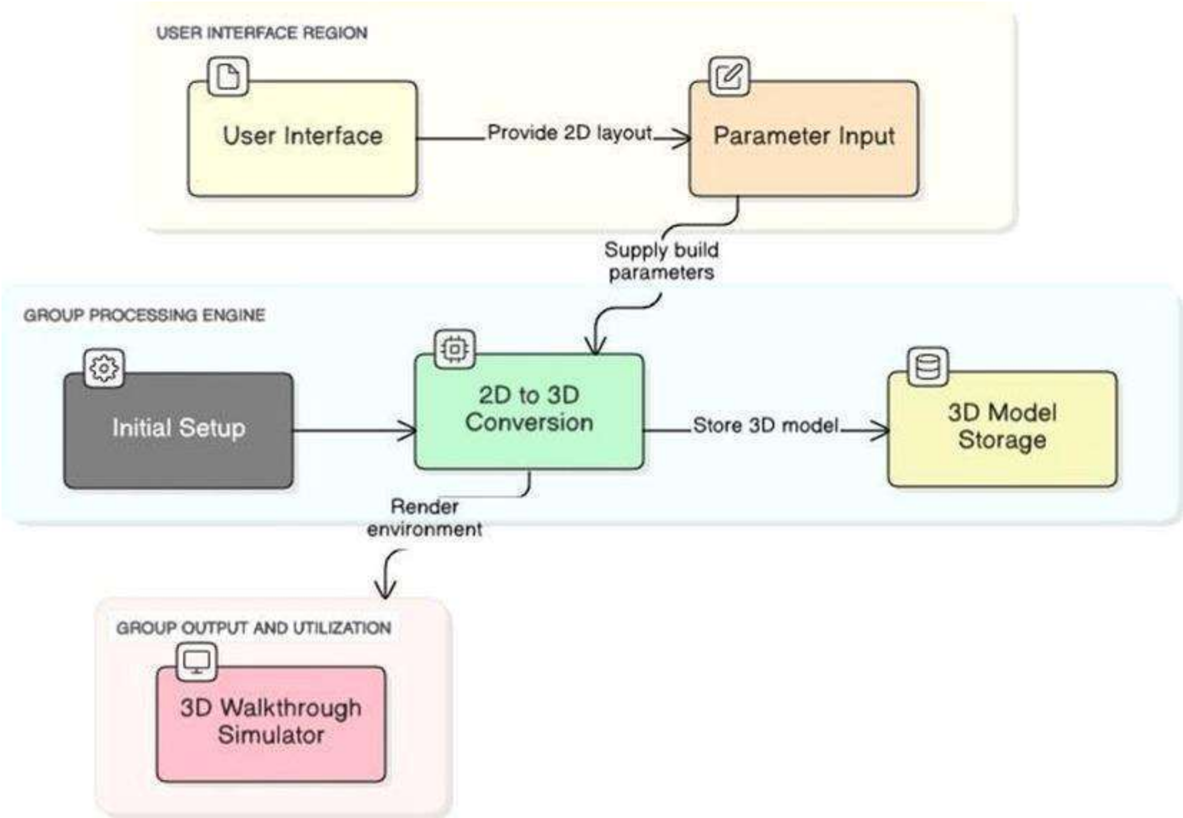

Fig. 1. Overall System Architecture

The first stage of the methodology focuses on the user interface and input. The user provides a 2D architectural layout in raster or vector format through the graphical user interface. Along with the blueprint, the user specifies essential build parameters such as wall height, scale, floor thickness, and rendering preferences through the parameter input module.

The second stage involves the group processing engine, which performs the core 2D-to-3D conversion. An initial setup module prepares the rendering environment by initializing coordinate systems, scale normalization, and scene settings. The 2D-to-3D conversion module processes the blueprint using image-processing techniques such as grayscale conversion, thresholding, contour detection, and geometric extraction to identify structural elements including walls, rooms, and openings [6], [14]. Procedural modeling techniques are then applied using the Blender Python API to extrude walls, generate floors and ceilings, and assemble a coherent 3D structure. Once generated, the 3D model is stored in the 3D model storage module for reuse, modification, or export in standard formats.

The final stage focuses on output and utilization, where the stored 3D model is loaded into the 3D walkthrough simulator. Lighting, camera paths, and collision handling are applied to enhance realism and usability. The walkthrough simulator allows users to visually inspect spatial layouts, assess navigation paths, and gain an understanding of the building structure, making the system suitable for architectural visualization, planning, and operational briefing applications.

The proposed system follows a fully automated, rule-based methodology to generate a three-dimensional (3D) multi-floor building model using DXF files as the only input. The methodology is designed to be robust, transparent, and independent of machine learning models [8], [15]. The complete workflow is implemented as a Blender add-on using Python scripting.

3.1 DXF Input and Preprocessing

The methodology begins by loading the input DXF file and accessing its model space to extract geometric entities. Only LINE entities are considered, as they represent wall centerlines in typical architectural blueprints. Degenerate or zero-length lines are ignored to prevent invalid geometry creation. Geometric bounds of the floor plan are calculated during preprocessing to determine the overall plan size. This information is later used for centering and automatic scaling of the generated 3D model.

3.2 Wall Geometry Construction

Wall generation is performed using a rule-based geometric approach. For each valid DXF line, a direction vector is computed and normalized. A perpendicular normal vector is derived and scaled according to half of the specified wall thickness. Using this offset, four vertices are generated to form a rectangular wall face corresponding to each line segment. All wall faces are constructed using mesh operations and merged into a single wall mesh. This approach ensures robustness even when the DXF file contains disconnected or unordered wall segments.

3.3 Scaling and Single-Floor 3D Extrusion

To maintain real-world proportionality, the generated floor plan is automatically scaled to a predefined target size. The scaling factor is computed using the maximum span of the DXF geometry. Once scaled, the wall mesh is extruded vertically to a fixed wall height, resulting in a complete 3D representation of a single-floor building model. This stage represents the core transformation from a 2D blueprint to a usable 3D architectural structure.

3.4 Floor Slab Generation

A floor slab is generated by tightly fitting a rectangular volume around the wall geometry using the bounding box dimensions. The slab is placed slightly below the wall base and extruded to a predefined slab thickness. This ensures structural completeness of each floor without relying on complex region detection algorithms.

3.5 Multi-Floor Integration

Multi-floor construction is achieved by duplicating the single-floor wall and slab models and translating them vertically along the Z-axis. Each floor is offset by a fixed floor-to-floor height value. This process ensures accurate alignment and vertical continuity across all floors. The number of floors is user-defined, allowing flexible generation of buildings with varying heights while maintaining uniform geometry and proportions.

3.6 Visualization Setup

For visualization and presentation purposes, basic materials are applied to the wall objects, and default lighting and camera configurations are added automatically. This enables immediate inspection and rendering of the generated multi-floor 3D building model within the 3D environment.

TABLE 1. Methodology modules and techniques used.

Module	Techniques Used
Input Processing	DXF parsing, geometric normalization
Preprocessing	Noise filtering, line validation
Feature Extraction	Wall line detection, boundary identification
Layout Validation	Geometry simplification, overlap removal

3D Model Generation	Blender Python API, mesh extrusion
Walkthrough Setup	Camera placement, lighting configuration
Export	.blend, .FBX, .glTF

TABLE 2. Step-wise algorithm description.

Step No.	Description
1	Load DXF blueprint and initialize Blender environment
2	Parse DXF entities and filter valid wall lines
3	Generate wall geometry using rule-based extrusion
4	Create floor slab based on bounding geometry
5	Construct single-floor 3D building model
6	Replicate and stack floors vertically
7	Apply materials, lighting, and export 3D model

IV. IMPLEMENTATION RESULTS

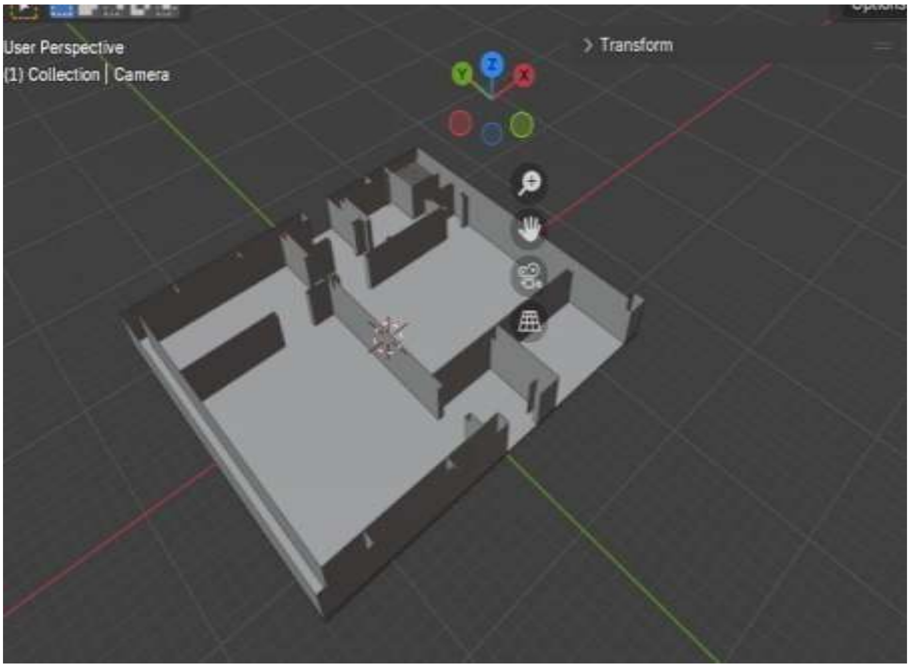

Fig. 2. Single-floor 3D building model generated from DXF blueprint.

Fig. 2 shows the automatically generated three-dimensional (3D) model of a single-floor building created from a 2D DXF architectural blueprint. The walls are constructed using a rule-based extrusion of line entities. The model demonstrates accurate spatial layout and wall alignment, serving as the base unit for subsequent multi-floor integration.

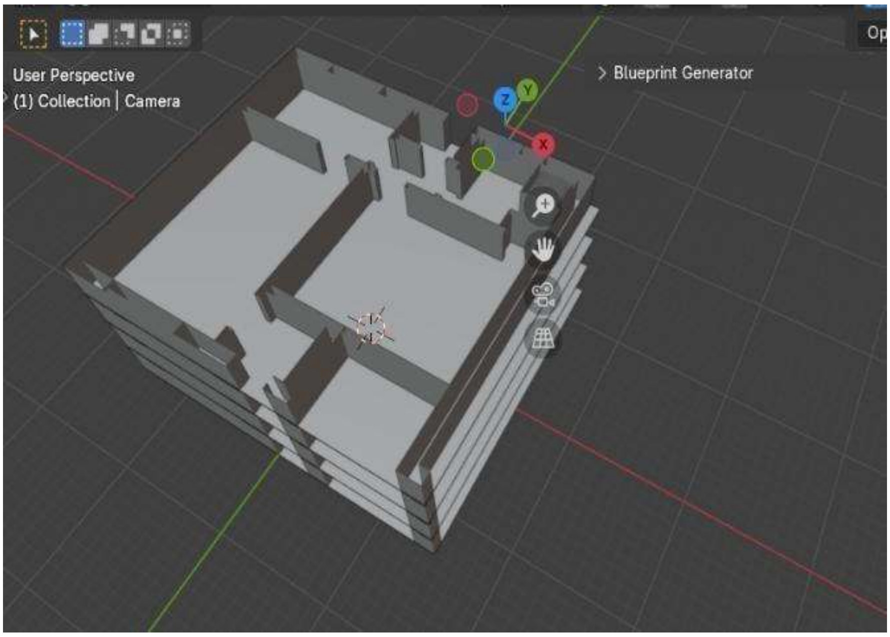

Fig. 3. Multi-floor 3D building model generated by vertical integration of single-floor units.

Fig. 3 illustrates the complete multi-floor three-dimensional (3D) building model obtained by stacking multiple single-floor models along the vertical axis. Each floor is positioned using a predefined floor-to-floor height, ensuring consistent alignment of walls and slabs across all levels.

V. CONCLUSION

This paper presented a rule-based methodology for automatic generation of multi-floor 3D building models using DXF files as the sole input. The proposed approach eliminates the need for machine learning models or proprietary BIM formats, ensuring transparency, reproducibility, and ease of implementation. The primary advantages of the system include full automation, robustness against imperfect DXF inputs, and flexible user control over architectural parameters such as wall dimensions and floor count. The methodology is particularly suitable for academic projects, rapid prototyping, and architectural visualization tasks. However, the system currently assumes linear wall representations and does not explicitly handle complex elements such as doors, windows, or curved geometries. Future work may focus on extending the rule set to support additional architectural features and improving semantic interpretation of DXF layers. The proposed approach demonstrates that reliable multi-floor 3D building models can be generated efficiently using purely geometric rules, making it a practical solution for digital construction workflows and computer-aided architectural modeling.

TABLE 3. comparative analysis

Parameter	**Existing Systems**	**PlanVista3D**
Input Type	Requires cleaned CAD/vector files or paired training datasets [14], [3]	Accepts scanned floorplans, raster images, and DXF files
Automation Level	Semi-automatic; requires manual preprocessing and corrections [2], [6]	Fully automatic blueprint-to-3D conversion with minimal user input
Core Technology	Deep Learning models, CAD tools, Cloud-based APIs [1]	OpenCV-based image processing + procedural modeling using Blender Python API
Processing Mode	Mostly cloud-dependent, dataset-driven [3]	Standalone local processing using Python and Blender
Offline Support	Requires active internet connection [1], [3]	Complete offline operation
Handling Real-World Blueprints	Poor performance on noisy, scanned, or low-quality plans [6], [12]	Robust preprocessing (thresholding, contour detection, noise removal)
Scalability	Dataset-dependent; retraining required for new layouts [8], [13]	Scales to large and complex buildings without retraining

REFERENCES

[1] Zhou J., Li X., Qi L., Yang M.-H., “Layout-Your-3D: Controllable and Precise 3D Generation with 2D Blueprint”, ICLR 2025, 2025.
[2] Saranya S., Varsha D., Sabarish S., “Conversion of 2D Blueprints into 3D Models”, International Journal of Information Technology and Research, 2025.
[3] H. T. Nguyen, Y. Chen, V. Voleti, V. Jampani, and H. Jiang, “HouseCrafter: Lifting Floorplans to 3D Scenes with 2D Diffusion Model,” arXiv preprint arXiv:2406.20077, 2024.
[4] S. Feist, L. J. de Sousa, L. Sanhudo, and J. P. Martins, “Automatic Reconstruction of 3D Models from 2D Drawings: A State-of-the-Art Review,” Eng—Advances in Engineering, vol. 5, no. 2, pp. 784-800, May 2024.
[5] S. Mamand and A. I. Abdulla, “2D to 3D Image Conversion Algorithms,” ITM Web of Conferences, vol. 64, 01010, 2024.
[6] L. Kratochvila, G. de Jong, M. Arkesteijn, S. Bilik, T. Zémčík, K. Horák, and J. S. Rellermeyer, “Multi-Unit Floor Plan Recognition and Reconstruction Using Improved Semantic Segmentation of Raster-Wise Floor Plans,” arXiv preprint arXiv:2408.01526, 2024.
[7] Barreiro, A. C., Trzeciakiewicz, M., Hilsmann, A., & Eisert, P. 2023. Automatic Reconstruction of Semantic 3D Models from 2D Floor Plans. Proceedings of MVA 2023.
[8] Bhalodia R., Elhabian S., Whitaker R., “DeepSSM: A Blueprint for Image- to-Shape Deep Learning Models”, Medical Image Analysis, 2024.
[9] Shinde P., Turate A., Mehta K., Nagpure R., “2D to 3D Dynamic Modeling in AR”, Proceedings of the 2022 International Conference on Augmented Reality, 2022.
[10] Bernabéu-Larena J., Hernández-Lamas P., Peinado-Santana S., “Public Works Heritage 3D Model Digitisation with Open-Source Tools”, Sustainability, 2021.
[11] Apollonio F.I., Fantini F., Garagnani S., Gaiani M., “A Photogrammetry- Based Workflow for the Accurate 3D Construction and Visualization of Museums Assets”, Remote Sensing, 2021.
[12] M. Vidanapathirana, Q. Wu, Y. Furukawa, A. X. Chang, and M. Savva, “Plan2Scene: Converting Floorplans to 3D Scenes,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2021.
[13] S. Park and H. Kim, “3DPlanNet: Generating 3D Models from 2D Floor Plan Images Using Ensemble Methods,” Electronics, vol. 10, no. 22, Article 2729, Nov. 2021.
[14] J. Zhu, H. Zhang, and Y. Wen, “A New Reconstruction Method for 3D Buildings from 2D Vector Floor Plan,” Computers, Materials & Continua, vol. 11, no. 3, 2021.
[15] Z. Kang, J. Yang, Z. Yang, and S. Cheng, “A Review of Techniques for 3D Reconstruction of Indoor Environments,” ISPRS International Journal of Geo-Information, vol. 9, no. 2020.