Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Deep Learning-Based BIM Automation: Advanced 3D Reconstruction with Material Integration for Indoor Elements

Mostafa Mahmoud [1*], Wu Chen [2], Mahmoud Adham [3], Yaxin Li [4]
[1, 2, 3, 4] *Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong*
[1, 3] *Public Works Department, Faculty of Engineering, Cairo University, Egypt*
[4] *Micro Dimension Technology Limited, Hong Kong*
**Corresponding Author Email: mostafa.mahmoud@connect.polyu.hk*

ARTICLE INFO

Received: 30 Dec 2024

Revised: 19 Feb 2025

Accepted: 27 Feb 2025

ABSTRACT

Building information modeling (BIM) models play a crucial role in reducing project expenses, supporting building design and renovation efforts, and enhancing the efficiency of building management. However, BIM models often emphasize geometric details while neglecting essential material attributes, which are vital for accurate facility assessments, budgeting estimation, and sustainable design practices. Current material recognition techniques typically exhibit limited accuracy and focus on a narrow range of materials, with an emphasis on 2D image detection rather than 3D reconstruction. This research introduces a novel scan-to-BIM framework that employs deep learning to automatically incorporate both geometric and material attributes from terrestrial laser scanner point cloud data. It offers a precise workflow for reconstructing structural and non-structural 3D BIM models, capturing geometry, position, quantity, shape, and orientation features. A novel material classification model based on point cloud characteristics is proposed. An integrated algorithm is also included in the Revit platform to enable automated 3D modeling of indoor components. The findings demonstrate the framework's effectiveness in achieving high geometric accuracy for structural elements while ensuring non-structural 3D BIM models retain key real-world characteristics. Furthermore, the material recognition model showed promising performance, with a point-based weighted average F1-score of 0.97 and an object-based weighted average accuracy of 94.74%.

Keywords: Building information modeling (BIM); scan-to-BIM; material classification; 3D modeling; point clouds; deep learning.

INTRODUCTION

Presently, indoor 3D reconstruction has been utilized efficiently in several areas, such as augmented and virtual reality, autonomous navigation in three-dimensional spaces, and improved oversight of construction projects [1] [2] [3]. As-built BIM models are crucial for minimizing the risk of expensive mistakes during construction and renovation processes; this can be achieved by converting point clouds into enhanced BIM models (scan-to-BIM) [4]. The importance of managing and operating buildings has increased due to their longer duration and higher costs compared to the initial construction phases [5]. Geometric information (such as dimensions) and attribute information (like materials) of BIM models are essential for activities such as replacements, repairs, and operation of buildings [6][7]. There is growing interest in automating BIM generation to reduce manual intervention, particularly for attribute data, as most current methods focus mainly on geometric aspects [8]. A BIM model with only geometric data is insufficient for assessing a facility's condition, as it neglects critical material information necessary for accurate modeling and cost estimation [9]. Integrating material details enhances quantification, prevents budget overruns, and supports asset management [10]. Deep learning techniques improve point cloud segmentation and material detection, further enhancing the scan-to-BIM process.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Available reconstruction methods lack a unified framework for automatically integrating BIM geometric modeling of structural and non-structural elements, as well as material attributes. Existing material recognition techniques often exhibit low classification accuracy and focus primarily on 2D RGB images, neglecting the precision of 3D reconstruction. BIM models typically require human assistance for geometric and attribute features, and geometric challenges, such as clustering limitations, hinder the distinction of independent non-structural objects and their varied shapes and orientations.

To address these challenges, this study presents an automated scan-to-BIM pipeline designed for creating as-built 3D BIM models. This innovative framework utilizes deep learning to incorporate both geometric and material considerations for structural and non-structural indoor elements. A precise workflow has been established to effectively generate 3D structural BIM models while also encompassing the modeling of non-structural elements by considering their geometry, position, quantity, type, and orientation. A novel deep learning material recognition technique that leverages point cloud geometric and intensity features is introduced. An automatic BIM modeling algorithm is developed in the Revit platform, considering geometric and attribute information. The paper is organized as follows: we review related work, detail the proposed methodology, and present experimental results, concluding with insights and future research recommendations.

RELATED WORKS

Current reconstruction techniques mainly focus on modeling essential structural components such as walls and floors to create 3D BIM representations of indoor spaces. This study emphasizes explicitly deep learning reconstruction and modeling methods. [11] presented an automated process for generating indoor BIM models from RGB-D data, using a neural network for semantic segmentation combined with 3D point clouds from a SLAM system. Though effective, manual adjustments may still be needed in complex or noisy environments. [12] proposed a framework for indoor point cloud reconstruction and BIM model automation but struggles with non-structural elements, relying on manual input for orientations and material identification. Recent advanced BIM modeling methods enable rebuilding non-structural interior elements like furniture. [13] developed a deep learning-based framework for reconstructing 3D models from point clouds in diverse room layouts, effectively generating 3D BIM models of furniture types. However, it focuses mainly on geometric features and requires manual input for material attributes.

Incorporating material data into scan-to-BIM elements enhances cost estimation, improves visualization, and supports maintenance, ultimately boosting performance. [14] proposed BIM-based material passports to document the material of buildings, leveraging materials and geometries extracted from radar images and point clouds. However, this method remains semi-automated, requiring human experts to manually input data for BIM model generation, including material details. [15] automated building object reconstruction using semantic segmentation but cannot reconstruct structural elements, requires manual interfering for point cloud matching, and is limited to five material classes. [16] utilized laser intensity variations for material classification but faced challenges due to distance and surface color affecting reflectance. To overcome this limitation, our proposed study utilizes point clouds with intensity data alongside geometric and color features in a newly developed deep neural network (DNN) model for material recognition.

METHODOLOGY

We propose a framework for generating 3D BIM models considering geometric and attribute information from scanned point clouds, as illustrated in Figure 1. The process begins with pre-processing the point cloud data through subsampling, filtering, and intensity feature calculation. A 3D deep learning algorithm performs instance segmentation to identify objects, with structural elements (like walls) reconstructed by detecting boundaries and openings and non-structural elements (like tables) analyzed for geometry and orientation. Our proposed DNN model identifies material types based on color, geometry, and intensity data. Finally, detailed 3D BIM models are automatically generated using a custom algorithm integrated into the Revit platform, streamlining the scan-to-BIM process.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

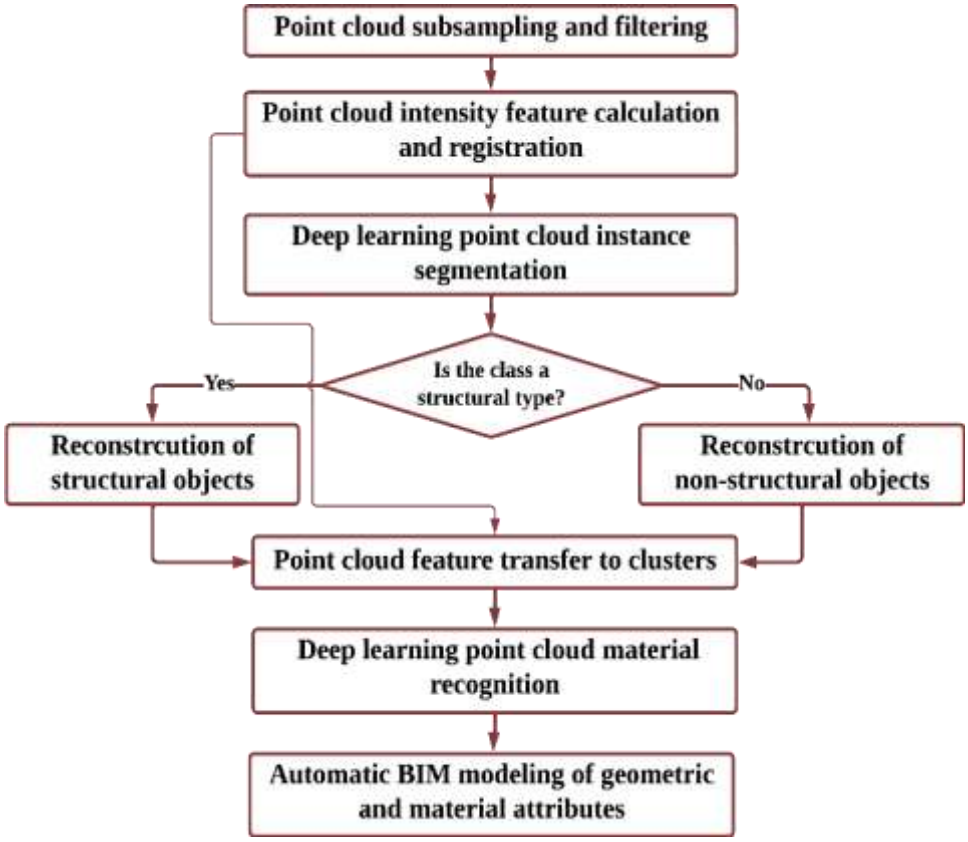


Figure 1. Proposed 3D reconstruction framework

Point Cloud Subsampling and Filtering

Raw scanned point cloud data is often dense, making processing resource-intensive and time-consuming. To improve efficiency, we employed subsampling and filtering methods. Our subsampling algorithm divides the domain into uniform grid cells, or voxels, each representing the mean value of points within it. The grid voxel size is a key input parameter. Additionally, to address remaining outliers, we applied a statistical outlier removal algorithm that eliminates points significantly farther from their neighbors than the average distance. This process requires two input parameters: the average distance and the number of neighbors, which are detailed in the results section.

Point Cloud Feature Calculation and Registration

The point clouds in this study include seven features: (x, y, z) are spatial coordinates, (r, g, b) are color values, and (i) is the intensity. The instance segmentation model uses the coordinates and color information, while reconstruction relies on the segmentation results. In our material recognition algorithm, intensity is crucial as it indicates the reflectivity of laser pulses from object surfaces. This intensity helps differentiate materials based on reflectivity, aiding object classification and surface characterization. This study recognizes that intensity values can be affected by surface texture, color, and object location. Therefore, we consider the features of intensity, color, and intensity for material classification. The intensity features include incident distance (d) and incident angle (θ), calculated using Equations (1-2).

$$d = \sqrt{((x - x_o)^2 + (y - y_o)^2 + (z - z_o)^2} \quad (1)$$

$$\theta = cos^{-1}(\frac{\vec{d}.\vec{n}}{|\vec{d}||\vec{n}|}) \quad (2)$$

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Here, x, y, z denote the coordinates of the point cloud, and x_o, y_o, z_o represent the origin point. The vector $\vec{n}$ is the surface normal, and $\vec{d}$ is the incident vector. The incident distance measures the distance from the laser scanner to each point. In contrast, the incident angle represents the angle between the incident vector and the surface normal at each point in the area of interest. The laser scanner is initially positioned at the origin of the coordinate system, but this only occurs during the first setup. For later scans, recalculating features is necessary before registering the point clouds, which involves aligning multiple scans into a unified coordinate system. The iterative closest point (ICP) [17] algorithm is used for registration while maintaining the intensity features. The registered point cloud data is then utilized for segmentation and reconstruction. In the material recognition phase, these additional intensity-based features are included.

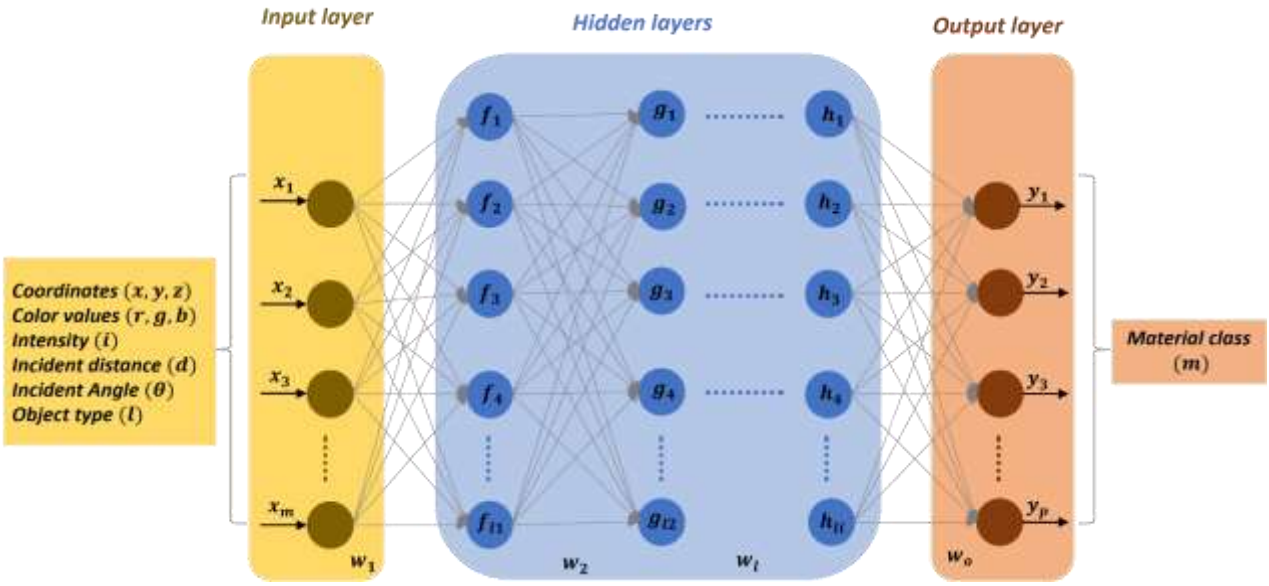


Figure 2. Proposed material recognition deep neural network.

Deep Learning Point Cloud Instance Segmentation

3D point cloud instance segmentation involves identifying each object within a 3D point cloud. The objective is to assign a unique label to each point, corresponding to specific objects like chairs or tables [18]. This process is crucial for object detection and recognition applications, and it improves augmented and virtual reality experiences by accurately segmenting real-world objects [19] [20]. The proposed instance segmentation model, TD3D [21], is a recent and precise approach that simplifies the complexity of literature methods by facilitating convolutional and efficient end-to-end training. The model functions in two main stages: proposal generation and proposal refinement. In the first stage, the model detects objects within the point cloud and generates initial bounding boxes as proposals. These proposals are then refined by a lightweight network in the second stage to produce the final instance masks. The input point cloud data consists of coordinates and raw color features, while the output includes semantic point cloud instances. The instance segmentation process generates 13 semantic cases based on the S3DIS dataset [21] used for training and testing, including structural elements like walls, ceilings, floors, columns, beams, doors, and windows, as well as non-structural items such as tables, bookcases, chairs, sofas, and boards. Points that did not fall within these categories were marked as clutter. The segmentation results facilitate the subsequent steps of reconstruction.

Reconstruction of Structural Objects

The reconstruction of indoor elements is divided into two categories based on instance segmentation classes: structural and non-structural components. The room boundary segment recognition detects wall lines and intersection points for modeling by analyzing points associated with wall and column classes from the instance segmentation results. It utilizes the iterative RANSAC algorithm to select point subsets, fit models, filter outliers, and refine parameters [22]. When multiple segments align with the same line, principal component analysis (PCA) [23]

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

is applied to distinguish them. This iterative method effectively estimates the parameters of plane surfaces while handling noise and outliers, producing reliable outcomes. The algorithm requires inputs such as an inlier threshold and the number of room lines for segment detection. To obtain intersections, the detected lines are organized in a sequence using logical sorting, which arranges them logically to help identify room corners where lines intersect. Detecting wall openings, such as windows and doors, is essential for realistic 3D building models.

The proposed method [13] identifies opening clusters, applies clustering for accurate separation, and eliminates incorrect clusters through analysis. Key attributes like center, length, and height are extracted using the least-squares technique, and the placement of opening points is refined to align with wall segments and maintain spatial relationships within BIM models.

Reconstruction of Non-Structural Objects

The process begins by selecting the relevant classes from instance segmentation results, such as chairs, tables, sofas, boards, and bookcases, to model indoor non-structural elements. Representing each class as a set of clusters enables straightforward geometric calculations. Bounding boxes are created around each cluster, capturing their coordinates and dimensions (length, width, and height) to ensure precise identification for later analysis. In Revit modeling, linking elements spatially offers advantages: windows and doors connect to walls, and furniture anchors to floors, preserving spatial relationships. Identifying the type and orientation of non-structural clusters is key to reconstructing non-structural objects. “Type” refers to an object’s shape (e.g., office or dining table), while “orientation” indicates its directional alignment.

Using a created reference point cloud library, clusters are matched to five main classes: chairs, tables, sofas, boards, and bookcases. An algorithm finds the best reference match for each cluster by minimizing the root mean squared error (RMSE). Once matched, the rotation angle is calculated to determine orientation. The algorithm begins with point cloud subsampling to reduce computational load and enhance efficiency. It then scales the test and reference point clouds by aligning their size ratios along the z-axis of their bounding boxes. Next, it calculates the translation needed to center both point clouds in the same position for easier alignment. For coarse registration, the fast point feature histogram (FPFH) [24] descriptor is used with the RANSAC algorithm. Finally, the ICP algorithm refines the alignment, selecting the point cloud with the lowest RMSE for each non-structural cluster.

Point Cloud Feature Transfer for Material Recognition

The instance segmentation output is formatted as $(x, y, z, (r, g, b)_{semantic})$ and used in reconstructing both structural and non-structural elements. However, for the material recognition workflow, it’s important to emphasize the original color and intensity features over instance segmentation colors. Using each point’s unique coordinates, we automatically replace the segmentation colors with raw color and intensity details, yielding the feature set. To fully leverage the instance segmentation labels, we integrate the segmentation semantic label into the material recognition dataset features. With 13 segmentation classes, each point is assigned a label (0 to 12) according to its instance cluster class. Consequently, the complete feature set for the material recognition model becomes $(x, y, z, r, g, b, i, d, \theta, l)$, covering each point's coordinates, original color values, intensity, distance, angle, and object labels.

Deep Learning Point Cloud Material Recognition

After preparing point cloud features, our proposed DNN model for material recognition, shown in Figure 2, includes an input layer, several hidden layers, dropout layers, and an output layer. The input layer processes normalized features from point cloud data, such as coordinates, color, intensity, distance, angle, and semantic labels. The hidden layers, decreasing in size from 512 to 64 neurons, employ batch normalization for training stability and ReLU activation functions to learn complex patterns. Dropout layers are included to prevent overfitting. The output layer, a dense layer with a SoftMax activation function, produces probability distributions for eight material classes: wood, metal, concrete, fabric, plasterboard, tile, plastic, and glass, as detailed in Section 4.1. Data augmentation techniques enhance the diversity of the training dataset, improving the model's generalization to unseen data.

$$Loss\,(y, \hat{y}) = -\sum_{i}^{n} p(y_i)\, log\,(p(\hat{y}_i)) \quad (3)$$

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Here, $p(y_i)$ and $p(\hat{y}_i)$ are the actual and predicted probabilities for i^{th} class and n is the class number. This project applies random drop, rotation, translation, and scaling to the input point cloud data to simulate real-world object orientation, position, and size variations, helping the model learn robust features. The Adam optimizer ensures fast convergence and effectively manages noisy gradients by adjusting learning rates for each parameter. The model uses sparse categorical cross-entropy as its loss function (Equation 3), which measures the cross-entropy between true labels and predicted probabilities, encouraging higher probabilities for correct classes.

Automatic BIM Modeling of Geometric and Material Attributes

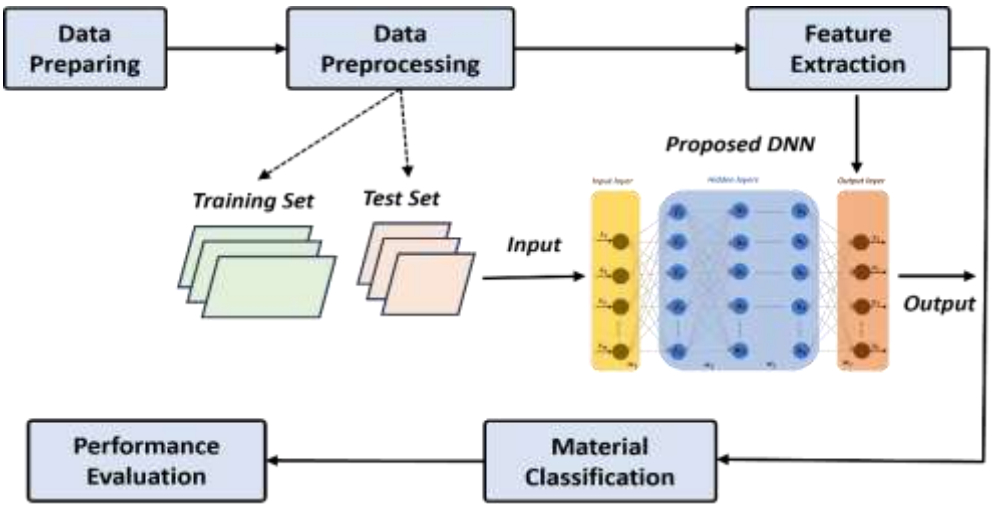


Figure 3. Workflow of proposed material recognition DNN.

The 3D BIM modeling phase combines all processed data to create final 3D models with geometric and attribute information. Structural elements include details such as room corners, opening centers, and dimensions, while non-structural elements encompass center location, dimensions, type, orientation, and material classes applicable to both categories. We use the Dynamo visual programming in the Revit platform to automate element creation, improving workflow efficiency and productivity. Customized scripts generate diverse BIM models through nodes, links, and Python scripts. To accurately represent indoor environments, we developed a BIM library of non-structural Revit families that align with the point cloud library, including attributes like name, type, and orientation.

We also updated the BIM material library to include eight types, wood, metal, concrete, fabric, plasterboard, tile, plastic, and glass, with texture, color, and reflectance properties. All BIM objects feature defined material parameters for seamless integration into Revit. The Dynamo algorithm for automatic BIM element modeling, illustrated in Figure 8 (left), first models structural objects like walls and floors using reconstruction data, then models non-structural objects like chairs and bookcases based on insertion points and dimensions and finally generates attribute material data identified by the DNN model.

RESULTS AND ANALYSIS

Workflow of Material Recognition and Dataset

The material recognition workflow includes key steps: data preparation, pre-processing for training and testing, applying the proposed DNN algorithm for feature learning and classification, and performance evaluation, as depicted in Figure 3. Due to the lack of 3D point cloud material classification datasets, we adapted existing data for our model. We utilized a large-scale public dataset [25] with accurate point clouds from various indoor environments. This dataset features point coordinates, raw colors, and intensity values and consists of separate scans. For training, we calculated incident distance and angle features, divided each scan into clusters, and segmented them into 13 classes to simulate instance segmentation results based on the S3DIS dataset. This process generated a feature set representing coordinates, color, intensity, intensity features, and object labels. Point cloud scans are subsampled for

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

efficiency and divided into training (80%) and testing (20%) datasets. We labeled the clusters with eight material categories: wood, metal, concrete, fabric, plasterboard, tile, plastic, and glass. The workflow concludes with testing the DNN material recognition model and evaluating its performance.

Training and Evaluation of Material Recognition Model

To evaluate the impact of point cloud features on material recognition, we incorporated all available calculated features ($x, y, z, r, g, b, i, d, \theta, l$). Model training took approximately 18 hours while testing on the dataset required only 5 minutes for 50 epochs. Figure 4 illustrates the model's performance, showing steady accuracy improvements up to 96.5% and a gradual reduction in loss until convergence, as expected for active deep learning models.

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$F1 = \frac{2 * P * R}{P + R} \tag{6}$$

Here, TP stands for true positives, FP represents false positives, and FN signifies false negatives.

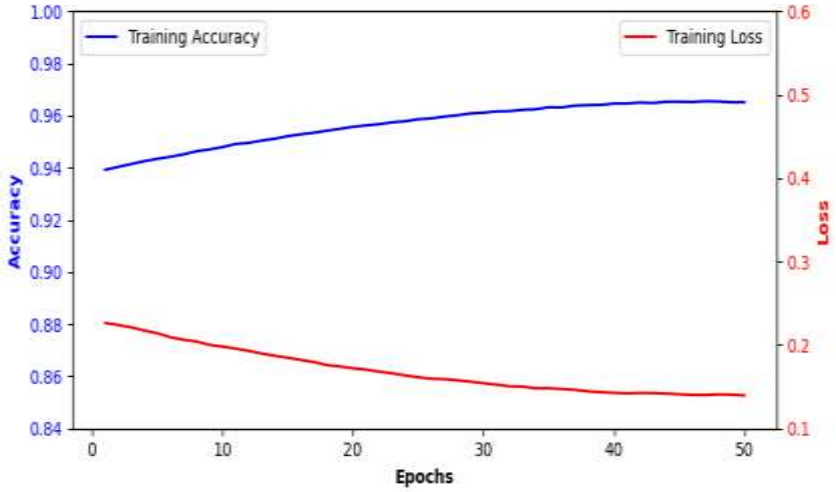


Figure 4. Workflow of proposed material recognition DNN.

To assess model performance during testing, precision (P), recall (R), and F1-score are calculated for all classes, as defined in Equations (4-6). These metrics demonstrate strong performance, with an average precision of 0.93, recall of 0.95, and F1-score of 0.94, underscoring the value of incorporating intensity and additional features for improved classification accuracy. Figure 5 provides a confusion matrix that shows the relationship between true and predicted labels, with darker cells indicating higher accuracy. This visualization confirms the model's effectiveness in identifying different material classes, with most achieving high classification accuracy between 92% and 98% for wood, metal, concrete, fabric, plasterboard, plastic, and glass. The tile class, however, has a slightly lower accuracy of 88%, likely due to limited training data, which makes it more challenging for the model to learn and generalize tile-specific features.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

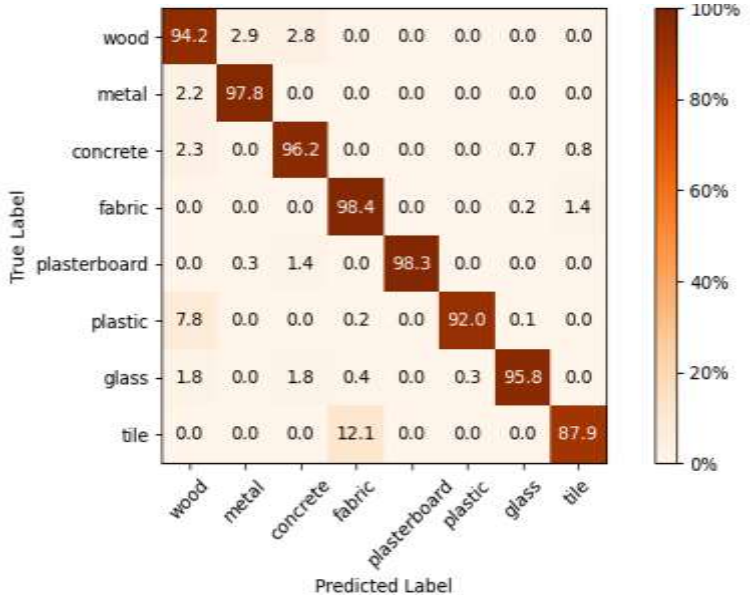


Figure 5. Confusion matrix for testing dataset.

Results and Assessment of Framework Reconstruction and Modeling

Dataset Description and Point Cloud Pre-processing

The evaluation dataset, shown in Figure 6(a), includes office room data from the public [25]. This laser-scanned data, featuring various furniture and materials, was pre-processed with point cloud filtering (1m average distance, 5 neighbors) and 1 cm voxel subsampling. Intensity features like incident angle and distance were computed, resulting in point cloud features with color, coordinates, and intensity data. Finally, the scans were then registered using the ICP algorithm, preserving these features.

Results of Point Cloud Instance Segmentation

Following registration, the point clouds, along with their coordinates and colors, were processed by the instance segmentation model. This model produced segmentation results that leveraged both semantic segmentation and object detection, enabling it to identify semantic classes and cluster points into distinct groups for each class. Figure 6(b) displays this instance segmentation results, where each cluster is represented in a unique color. The segmented elements were categorized into two primary groups: structural and non-structural. Structural elements included walls, floors, ceilings, columns, beams, windows, and doors, while non-structural elements comprised items like chairs, bookcases, sofas, tables, and boards.

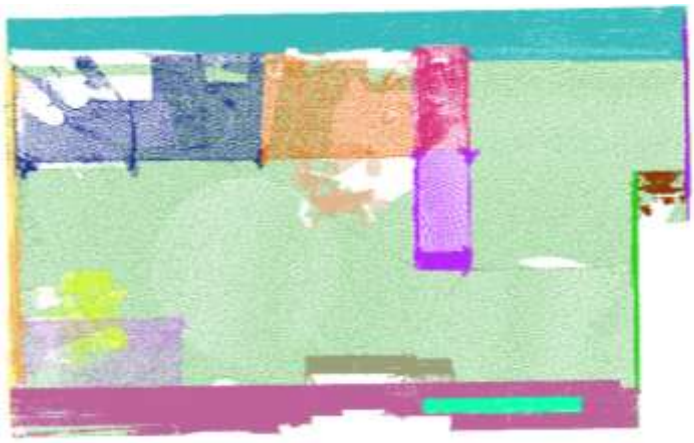
Results of Point Cloud Structural Reconstruction

After segmentation, the structural elements are reconstructed by defining both room boundaries and opening features. This includes detecting change points in rooms and extracting features from clusters related to openings. The process begins by identifying floor and ceiling levels using wall and column classes, employing iterative RANSAC and PCA algorithms. Detected boundary lines are sorted and intersected to extract room corners, which are used in the 3D BIM modeling stage.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

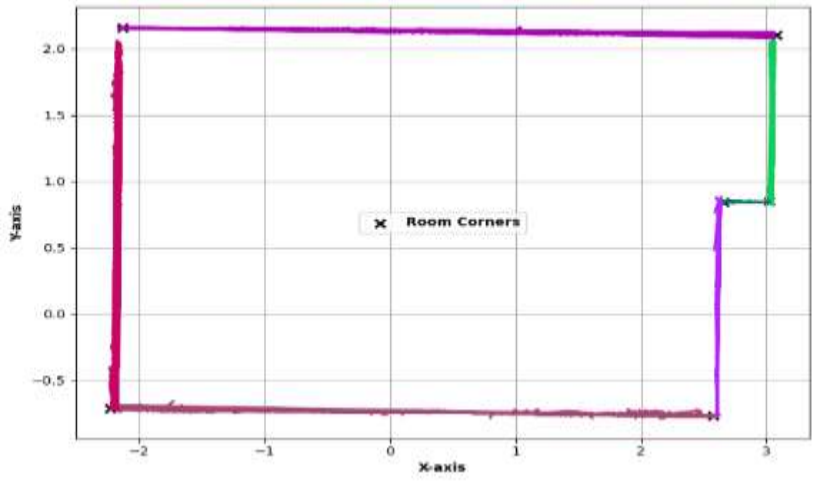
(a)



(b)

Figure 6. Evaluation of dataset: (a) raw data and (b) instance segmentation results.

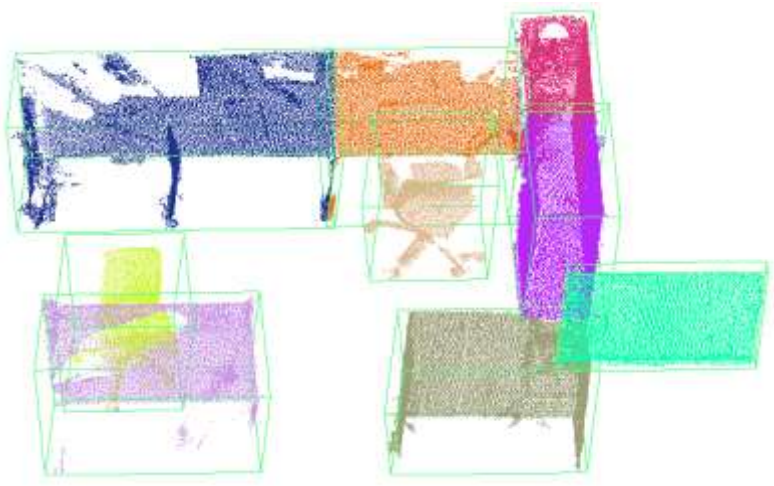
Figure 7(a) shows the room boundary segments with colored wall lines and room corners marked by black crosses. Segmented clusters for doors and windows are used to identify opening elements for accurate 3D modeling. The DBSCAN algorithm prevents the merging of connected clusters. Erroneous clusters are removed through analysis, and the least squares method determines the center, length, and height of each cluster. Detected opening points are refined to ensure alignment with walls, achieving 100% detection accuracy with two openings identified. These results confirm that the detected elements accurately represent actual wall surfaces.



Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

(a)



(b)

Figure 7. Reconstruction results: (a) structural boundary detection and (b) bounding boxes for non-structural elements

Results of Point Cloud Non-Structural Reconstruction

This step focuses on reconstructing non-structural elements like tables, bookcases, chairs, sofas, and boards, considering their counts, positions, dimensions, shapes, and orientations. Following instance segmentation, non-structural clusters are extracted with 100% accuracy, detecting 9 objects. Geometric bounding boxes predict their centers and dimensions (width, depth, height), as shown in Figure 7(b). The floor class is included to ensure accurate modeling and maintain spatial relationships. Our algorithm identifies the type and orientation of non-structural clusters by comparing their point clouds to a library of objects. A 2 cm voxel size is used for subsampling to reduce computation costs while focusing on key points. Scaling and translation are applied, followed by FPFH and ICP techniques for registration. The object with the lowest RMSE is selected, and its shape and orientation are considered for 3D modeling. The case study achieved an average matching RMSE of 0.038 m across different non-structural classes.

Results of Point Cloud Material Recognition

After reconstructing structural and non-structural objects, predicting material attributes is crucial for comprehensive BIM modeling. The instance segmentation process provides coordinates and semantic colors for both structural and non-structural elements. For material recognition, raw color, and features are replaced with segmentation colors based on unique coordinates, and the semantic class label for each cluster is utilized. This creates a complete feature set, including coordinates, colors, intensity features, and semantic labels, which are used in a deep-learning model for material recognition. The proposed DNN model predicts material classes like wood, metal, fabric, glass, tile, plasterboard, concrete, and plastic, determining the final material by the category with the most detected points. Table I presents classification results based on point-level predictions, comparing predicted labels with ground truth and showing high precision, recall, and F1 scores across materials. Metal achieved an F1 score of 0.88, while wood scored 0.86; concrete, fabric, plasterboard, and plastic achieved perfect scores. Overall weighted averages were 0.98 for precision and 0.97 for recall and F1-score, demonstrating the model's effectiveness in predicting material attributes for integrated 3D BIM modeling.

Table 1. Evaluation Of Point-Based Material Recognition.

Class	Precision	Recall	F1-score	Instances
Wood	0.75	1.00	0.86	49010

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Metal	1.00	0.78	0.88	75811
Concrete	1.00	1.00	1.00	243277
Fabric	1.00	1.00	1.00	80023
Plasterboard	1.00	1.00	1.00	89953
Plastic	1.00	1.00	1.00	10902
Weighted avg.	0.98	0.97	0.97	-

Results of Automatic 3D BIM Geometric and Attributes Modeling

Our custom Dynamo algorithm, integrated with Revit, automates the parametric BIM modeling of elements using processed geometric and material data from structural and non-structural objects. The script models structural elements based on predicted coordinates then adjusts dimensions and assigns types to non-structural elements. It reads material types from the BIM material library and automatically applies the matched materials to the BIM objects for accurate integration. As shown in Figure 8, the process highlights the Dynamo algorithm in the Revit, with snapshots of the generated 3D BIM models. These models accurately represent structural components and include non-structural elements, considering each object's location, size, shape, and orientation. The approach effectively assigns material properties to all components, demonstrating the algorithm's capability to convert point cloud data into detailed parametric models.

To evaluate modeling dimensional accuracy, we compared the predicted perimeter (16.13 m) to the actual perimeter (16.18 m), resulting in a length error of 0.05 m. The accuracy, calculated as 100% minus the error ratio, is 99.70%, indicating strong alignment between the model's dimensions and the reference measurements. The automatically generated non-structural 3D BIM models accurately reflect unit counts, positioning, dimensions, model types, and orientations, showcasing the framework's capability to reconstruct geometric and attribute information, as seen in Figure 8. Table II evaluates the accuracy of non-structural model types and orientations, revealing perfect shape accuracy of 100%, with all predicted shapes matching actual objects. The overall orientation accuracy is 89%, with errors primarily due to the detected rotation of one bookcase, requiring minor manual adjustments, as shown in Figure 9. These results highlight the model's effectiveness in accurately classifying BIM object types and orientations.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

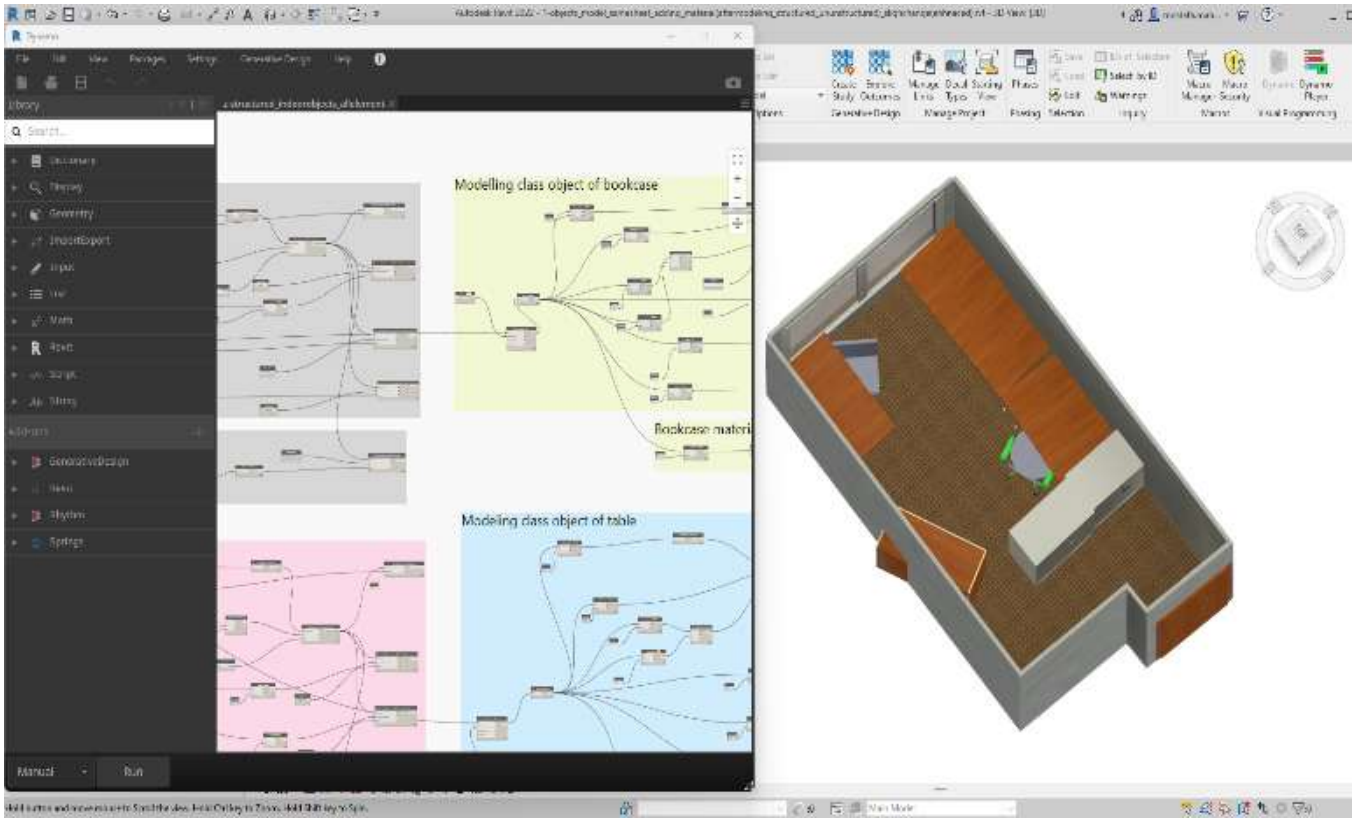


Figure 8. BIM parametric dynamo Revit algorithm (left) and automatically generated 3D models (right).

Figure 8 shows that the generated BIM models are geometrically modeled with their predicted materials. Table III presents the material recognition accuracy for each object class, with the model assigning a material category based on the predominant material detected among points. The board class, made of wood, achieved 100% accuracy, as did the ceiling, chair, door, floor, table, and wall classes, with their materials accurately identified. The bookcase class had a lower accuracy of 67%, with two of three metal bookcases correctly classified and one misclassified as wood.

Table 2. Accuracy Evaluation of Non-Structural Bim Object Type and Orientation.

Class	Unchanged object type count /actual count	Unchanged object orientation count /actual count
Bookcase	3/3	2/3
Board	1/1	1/1
Table	3/3	3/3
Chair	2/2	2/2
Total	9/9	7/9
Accuracy (%)	100	89

Overall, the model's weighted average accuracy across all object classes is 94.74%, indicating strong performance in material classification. The finalized 3D BIM models for the case study are shown in Figure 9, reflecting a minor correction of the misclassified bookcase material from wood to metal. The models include material tags: W for wood, C for concrete, P for plastic, M for metal, F for fabric, L for plasterboard, G for glass, and T for tile. In summary, our proposed automated framework effectively manages both structural and non-structural components, integrating geometric and material properties into the Revit platform.

Journal of Information Systems Engineering and Management
2025, 10(34s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Table 2. Evaluation Of Object-Based Material Recognition

Object Class	Actual Objects	Actual Material	Correctly Predicted	Predicted Material	Accuracy (%)
Board	1	Wood	1	Wood	100
Bookcase	3	Metal	2	Wood	67
				Metal	
Ceiling	1	Plasterboard	1	Plasterboard	100
Chair	2	Plastic	2	Plastic	100
Door	1	Wood	1	Wood	100
Floor	1	Fabric	1	Fabric	100
Table	3	Wood	3	Wood	100
Wall	6	Concrete	6	Concrete	100
Window	1	Metal	1	Wood	100
Weight. Avg.	-	-	-	-	94.74

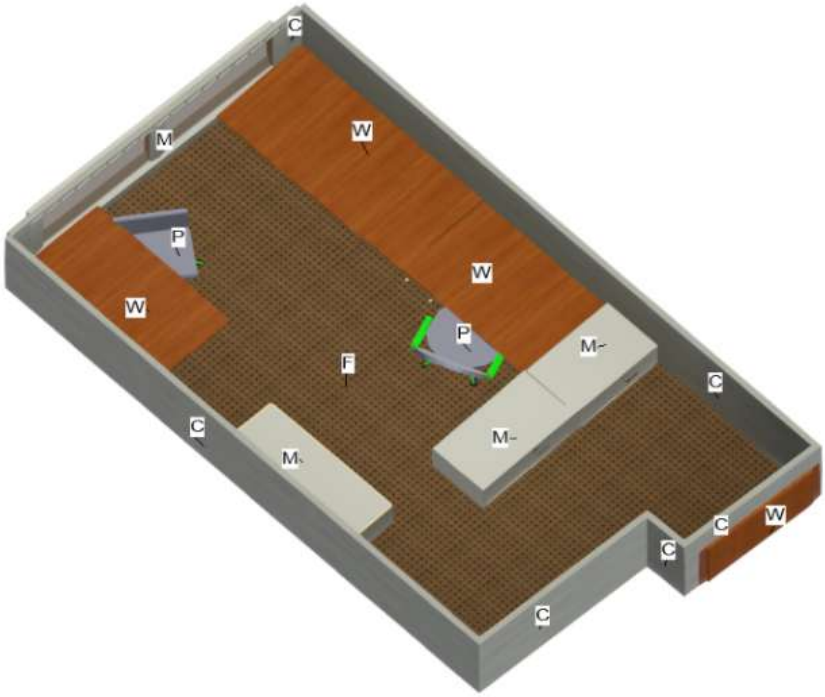


Figure 9. Final 3D BIM models of structural and non-structural components with geometric and material data.

SUMMARY AND CONCLUSION

Accurate as-built BIM models are essential in architecture and construction for cost reduction and effective future planning. During operation and maintenance, managing attribute data is crucial for maintaining building performance, as geometric information alone is insufficient without key parameters like material details. Material information enhances BIM accuracy, cost estimation, energy analysis, and asset management. Current

reconstruction methods lack a unified framework for integrating geometric modeling of structural and non-structural elements with material properties. Many materials recognition techniques struggle with low accuracy and focus on 2D detection rather than 3D reconstruction, complicating automation due to the varied shapes and orientations of non-structural objects.

This study presents an automated scan-to-BIM framework for creating as-built 3D BIM models. Utilizing deep learning, the framework integrates geometric and material attributes for both structural and non-structural elements, producing accurate models that capture geometry, position, quantity, shape, orientation, and material properties. A novel deep-learning technique for material classification is also introduced. The proposed method facilitates automatic 3D BIM modeling through an algorithm implemented in the Revit platform. The results demonstrate that the method achieves highly accurate models for structural elements and reflects 100% accuracy for non-structural object types, with 89% accuracy for orientations. The material recognition DNN records a weighted average F1-score of 0.97 and an object-based accuracy of 94.74%. Future work aims to enhance the framework's ability to manage mixed-material objects and improve material recognition by integrating thermal imaging with point cloud features. These advancements will increase the framework's versatility and robustness across various environments.

Acknowledgement:

This study was supported by funding from the Hong Kong Research Grants Council (RGC) through the Research Impact Fund, project number R5011-23.

Conflict of Interest:

The authors declare no conflict of interest.

REFERENCES

[1] M. A. Rabbou, M. Mahmoud, and A. El Shazly, "Performance Evaluation of Single Frequency PPP using Smartphone ' s Raw GNSS Observations," vol. 43, no. 2, pp. 77–93, 2021, [Online]. Available: https://www.azharcermjournal.com/CERMF2104/P21-04-06.pdf

[2] M. Mahmoud, M. Abd Rabbou, and A. El Shazly, "Land Vehicle Navigation Using Low-Cost Integrated Smartphone GNSS Mems and Map Matching Technique," *Artif. Satell.*, vol. 57, no. 3, pp. 138–157, 2022, doi: 10.2478/arsa-2022-0007.

[3] M. Mahmoud, "Evaluation of Smartphone GNSS Precise Point Positioning Integrated With MEMS and Road Map Matching Technique Supporting Land Vehicle Navigation System," Cairo University, 2022.

[4] G. Pintore, C. Mura, F. Ganovelli, L. Fuentes-Perez, R. Pajarola, and E. Gobbetti, "State-of-the-art in automatic 3D reconstruction of structured indoor environments," *Comput. Graph. Forum*, vol. 39, no. 2, pp. 667–699, 2020, doi: 10.1111/cgf.14021.

[5] Y. Li, W. Li, W. Darwish, S. Tang, Y. Hu, and W. Chen, "Improving plane fitting accuracy with rigorous error models of structured light-based RGB-D sensors," *Remote Sens.*, vol. 12, no. 2, pp. 1–17, 2020, doi: 10.3390/rs12020320.

[6] D. K. Abideen, A. Yunusa-Kaltungo, P. Manu, and C. Cheung, "A systematic review of the extent to which BIM is integrated into operation and maintenance," *Sustainability*, vol. 14, no. 14, p. 8692, 2022, doi: doi.org/10.3390/su14148692.

[7] A. Adham, Mahmoud , Mohamed, Mohamed G. , El Shazly, "Railway Tracks Detection of Railways Based on Computer Vision Technique and Gnss Data," in *ICCSTE'20*, 2020, pp. 1–8. doi: 10.11159/iccste20.269.

[8] J. Jung *et al.*, "Productive modeling for development of as-built BIM of existing indoor structures," *Autom. Constr.*, vol. 42, pp. 68–77, 2014, doi: 10.1016/j.autcon.2014.02.021.

[9] R. Volk, J. Stengel, and F. Schultmann, "Building Information Modeling (BIM) for existing buildings—Literature review and future needs," *Autom. Constr.*, vol. 38, pp. 109–127, 2014, doi: 10.1016/j.autcon.2013.10.023.

[10] A. Zabin1a, B. Khalil2b, T. Ali1c, J. A. Abdalla, and A. Elaksher3d, "A semi-automated method for integrating textural and material data into as-built BIM using TIS," *Adv. Comput. Des.*, vol. 5, no. 2, pp. 127–146, 2020, doi: 10.12989/acd.2020.5.2.127.

Research Article

[11] Y. Li, W. Li, S. Tang, W. Darwish, Y. Hu, and W. Chen, "Automatic indoor as-built building information models generation by using low-cost RGB-D sensors," *Sensors (Switzerland)*, vol. 20, no. 1, 2020, doi: 10.3390/s20010293.

[12] M. Mahmoud, W. Chen, Y. Yang, and Y. Li, "Automated BIM Generation for Large-Scale Indoor Complex Environments Based on Deep Learning," *Autom. Constr.*, vol. 162, no. June, p. 105376, 2024, doi: 10.1016/j.autcon.2024.105376.

[13] M. Mahmoud, W. Chen, Y. Yang, T. Liu, and Y. Li, "Leveraging Deep Learning for Automated Reconstruction of Indoor Unstructured Elements in Scan-to-BIM," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch.*, vol. 48, no. 1, pp. 479–486, 2024, doi: 10.5194/isprs-archives-XLVIII-1-2024-479-2024.

[14] M. Honic, I. Kovacic, I. Gilmutdinov, and M. Wimmer, "Scan to BIM for the Semi-Automated Generation of a Material Passport for an Existing Building," in *37th CIB W78 Information Technology for Construction Conference (CIB W78)*, 2020, pp. 338–346. doi: 10.46421/2706-6568.37.2020.paper024.

[15] S. Kim, K. Jeong, T. Hong, J. Lee, and J. Lee, "Deep Learning–Based Automated Generation of Material Data with Object–Space Relationships for Scan to BIM," *J. Manag. Eng.*, vol. 39, no. 3, p. 4023004, 2023, doi: 10.1061/jmenea.meeng-5143.

[16] S. Tao, Q. Guo, S. Xu, Y. Su, Y. Li, and F. Wu, "A geometric method for wood-leaf separation using terrestrial and simulated lidar data," *Photogramm. Eng. Remote Sens.*, vol. 81, no. 10, pp. 767–776, 2015, doi: 10.14358/PERS.81.10.767.

[17] J. Zhang, Y. Yao, and B. Deng, "Fast and robust iterative closest point," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3450–3466, 2021, doi: 10.1109/TPAMI.2021.3054619.

[18] A. Mahmoud, M. G. Mohamed, and A. El Shazly, "Low-cost framework for 3D reconstruction and track detection of the railway network using video data," *Egypt. J. Remote Sens. Sp. Sci.*, vol. 25, no. 4, pp. 1001–1012, 2022, doi: 10.1016/j.ejrs.2022.11.001.

[19] M. Mahmoud *et al.*, "Automated Scan-to-BIM: A Deep Learning-Based Framework for Indoor Environments with Complex Furniture Elements," *J. Build. Eng.*, no. 112596, pp. 1–24, 2025, doi: 10.1016/j.jobe.2025.112596.

[20] M. Adham, W. Chen, Y. Li, and T. Liu, "Towards Robust Global VINS : Innovative Semantic- Aware and Multi-Level Geometric Constraints Approach for Dynamic Feature Filtering in Urban Environments," IEEE, 2024. doi: 10.1109/TIV.2024.3487593.

[21] M. Kolodiazhnyi, A. Vorontsova, A. Konushin, and D. Rukhovich, "Top-down beats bottom-up in 3d instance segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 3566–3574. [Online]. Available: https://openaccess.thecvf.com/content/WACV2024/papers/Kolodiazhnyi_Top-Down_%0ABeats_Bottom-Up_in_3D_Instance_Segmentation_WACV_2024_paper.pdf %0A.

[22] M. Mahmoud, Y. LI, M. Adham, and W. CHEN, "Automated Material-Aware BIM Generation : Deep Learning for Comprehensive Reconstruction of Indoor Elements," *Autom. Constr.*, pp. 1–35, 2025.

[23] C. Labrín and F. Urdinez, "Principal component analysis," in *R for Political Data Science*, vol. 1, no. 1, Chapman and Hall/CRC, 2020, pp. 375–393. doi: 10.1016/0169-7439(87)80084-9.

[24] W. Xie, Z. Zhang, Y. Wang, Y. Zhang, and L. Zhu, "The New Fast Point Feature Histograms Algorithm Based on Adaptive Selection," *J. Appl. Sci. Eng.*, vol. 23, no. 2, pp. 225–232, 2020, doi: 10.6180/jase.202006_23(2).0006.

[25] "UZH Dataset." Accessed: Aug. 19, 2024. [Online]. Available: https://www.ifi.uzh.ch/en/vmml/research/datasets.html