Automated 3D Building Model Reconstruction using Orthophotos and Point Clouds

Dinh Minh Bui[1], Hyunsoo Kim[2], Junhee Youn[3], Changjae Kim[4]

[1]Dept. of Civil Engineering and Environment, Myongji University, Korea – datb864@gmail.com
[2]Dept. of Civil Engineering and Environment, Myongji University, Korea – ae1996@naver.com
[3]Dept. of Future Smart Construction, Korea Institute of Civil Engineering and Building Technology – younj@kict.re.kr
[4]Dept. of Civil Engineering and Environment, Myongji University, Korea – cjkim@mju.ac.kr

Keywords: 3D Building Reconstruction, Lines Detection, Clustering Point Cloud, Modified U-Net, Mobile Line Segment Detector

Abstract

The accurate reconstruction of 3D building models is essential for urban planning and smart city applications. This study introduces an automated workflow integrating Unmanned Aerial Vehicle (UAV)-derived orthophotos and point clouds to enhance reconstruction accuracy. A deep learning-based tree segmentation model filters non-building objects, while the Cloth Simulation Filter (CSF) separates ground and non-ground points. Clustering techniques isolate building structures, followed by Mobile Line Segment Detector (LSD)-based roof edge detection and refinement. The extracted roof edges are then combined with height attributes to generate 3D bounding boxes. Experiments on UAV data from Suseo, South Korea, show that this approach reconstructs detailed and realistic 3D models, achieving high precision and recall measures. By integrating deep learning, clustering, and geometric analysis, this study provides a scalable and efficient solution for urban modeling.

1. Introduction

The increasing complexity of urban environments calls for accurate 3D models to support urban planning, environmental monitoring, and disaster management. UAVs have emerged as a powerful tool for capturing high-resolution geospatial data, enabling efficient modeling of urban features (Zhang et al., 2022). Orthophotos and point clouds derived from UAV imagery provide critical datasets for the abovementioned tasks (Gómez & Téllez, 2020).

By integrating orthophotos and point clouds, this study leverages the strengths of both datasets. Orthophotos offer high-resolution, georeferenced imagery that captures detailed surface features like rooftops and vegetation, aiding classification and segmentation. Meanwhile, point clouds provide 3D structural information, mitigating geometric distortions and enabling accurate height estimations. This complementary integration ensures a robust and detailed approach to urban reconstruction.

This study proposes an integrated workflow for 3D building reconstruction using UAV-derived data. Deep learning is applied to orthophotos for tree segmentation, while the CSF algorithm is used to separate ground and non-ground points from the original point cloud (Zhao et al., 2016). Clustering techniques then filter low-elevation points, leaving key structures like buildings. By combining building-specific point clouds and orthophotos, roof edge lines are detected, refined through clustering, and extended into bounding boxes enriched with height attributes from point cloud data. These bounding boxes form the basis for generating 3D building models.

This approach demonstrates the synergy of UAV-derived data, deep learning, and clustering methods, addressing challenges in urban modeling while contributing to advancements in geospatial analysis.

2. Methodology

The workflow, shown in Figure 1, illustrates the methodology proposed for 3D building reconstruction in this study. As a pre-processing step, point clouds and orthomosaic imagery are generated from UAV images using commercial photogrammetric software. Afterwards, as a first process of non-building object filtering step, ground filtering is applied to the point clouds to separate ground and non-ground points. Meanwhile, tree extraction step is carried out on the orthomosaic imagery. The filtered data enable the identification of building points and the extraction of 2D building boundaries. Afterwards, these datasets are integrated and used to detect roof edges and extract building candidates, forming the foundation for accurate 3D building reconstruction. The methodology highlights the integration of UAV-derived data, point cloud processing, and orthophoto analysis to achieve robust and reliable results. The detailed explanations of all the procedures in the proposed methodology are provided in the following subsections.

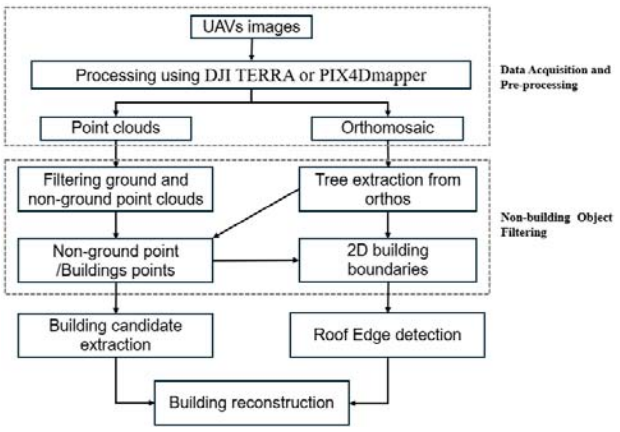


Figure 1. Overview of the proposed methodology

2.1 Data Acquisition and Pre-processing

UAV imagery is systematically captured over the designated study area to provide detailed visual data. This data acquisition phase involves taking a series of high-resolution aerial photographs from multiple angles to ensure comprehensive coverage. The images are then processed using advanced photogrammetric software, such as DJI TERRA or PIX4Dmapper (Figure 2). These software tools analyze the images and use techniques like Structure from Motion (SfM) and Multi-View Stereo (MVS) to reconstruct spatial information.

Figure 2. UAV images and softwares for generating orthophotos and pointclouds

The photogrammetric processing generates two primary outputs: orthophotos and point clouds. Orthophotos are geometrically accurate 2D images that maintain a uniform scale and eliminate distortions caused by terrain and camera tilt. These high-resolution images are valuable for tasks requiring accurate spatial measurements and detailed mapping. On the other hand, point clouds represent the 3D structure of the environment by mapping millions of points in space, capturing the precise geometry of buildings, vegetation, and other surface features. These outputs provide a robust foundation for various applications, such as urban planning, environmental monitoring, and 3D modeling.

2.2 Non-building Object Filtering

The overview of the object filtering process applied to non-ground point clouds for isolating building structures is shown in Figure 3. The workflow combines data from point clouds and orthomosaic imagery to ensure accurate separation of buildings from other objects.

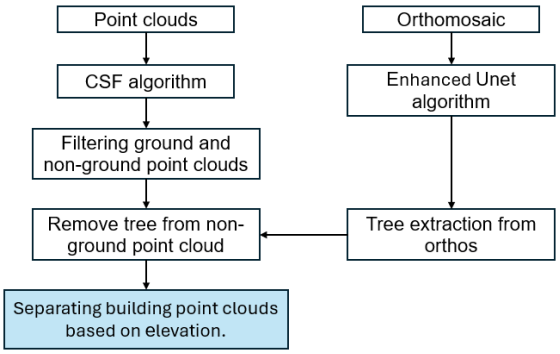


Figure 3. Overview of object filtering in non-ground point clouds

Initially, the CSF algorithm is used to filter the ground and non-ground points from the raw point cloud data. Simultaneously, an enhanced U-Net model processes the orthomosaic to extract tree regions. These tree regions are then used to remove corresponding points in the non-ground point cloud. Following this step, a filtering process based on elevation thresholds is applied to the remaining points in the non-ground point cloud, effectively isolating building structures.

More specifically, CSF algorithm is utilized to separate the original point cloud into ground and non-ground points. The non-ground points include features such as buildings, trees, vehicles, people, and other non-ground objects. The CSF algorithm simulates a cloth over the point cloud, identifying ground points by detecting the lowest elevations. The filtering parameters are adjusted to accommodate varying terrain conditions, ensuring an accurate separation of buildings from non-ground elements.

At this stage, one should note that the U-Net model utilized in this research has been modified to improve tree segmentation from orthophoto images as follows: 1) Layer Normalization is added after each convolution. 2) SeparableConv2D replaces standard convolutions to reduce computational complexity. 3) The Dice coefficient is used to address class imbalance. 4) 1x1 convolutions during up sampling improve efficiency without loss of spatial detail (Figure 4). The example of tree segmentaiton result using the enhanced U-Net model is shown in Figure 5.

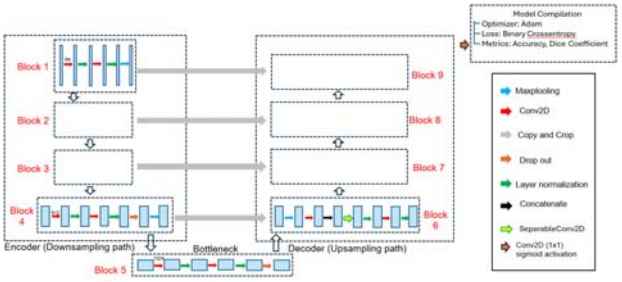


Figure 4 . Enhanced U-Net model framework

Figure 5. Tree segmentation result using U-Net model

Even though the CSF and tree segmentation algorithms are applied to separate building point clouds from the original point data, we still have some outliers from vehicles, people, and other non-ground objects. Hence, the height-based filter should be carried out to eliminate such outliers except for buildings. Relative height information of the building structures is derived

from the neighboring ground points. Based on this height information, height-based filter is applied.

2.3 Building Candidate Extraction

This step focuses on identifying the locations of building candidates based on the outcomes of subsection 2.2. Using orthophoto-based geospatial information and point clustering techniques, distinct building candidates are identified as separate ones. These building candidates include both X, Y, Z point and imagery information. Hence, textural and structural information of building candidates can be utilized for building reconstruction in this research.

Since the outcomes of subsection 2.2 contain multiple buildings without clear separation, a clustering technique is applied to identify individual structures. In this step, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is employed due to its ability to handle varying building sizes and densities without requiring prior knowledge of the number of clusters (Figure 6).

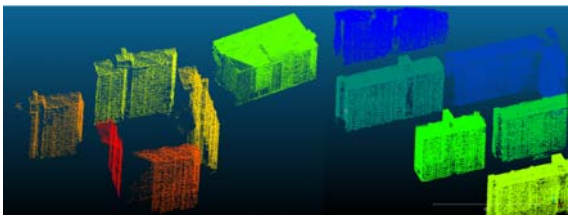


Figure 6. Building candidates clustering result

As long as the UAV-derived orthophoto and point cloud share the same coordinate system, each point within a building cluster can be mapped to its corresponding location in the orthophoto. This allows the extraction of precise building images directly from the orthophoto.
The transformation from 3D point cloud coordinates (x, y, z) to 2D image coordinates (x_{img}, y_{img}) is computed as shown in equation (1).

$$(x_{\text{img}}, y_{\text{img}}) = \left(\frac{x - x_{\min}}{\text{GSD}}, \frac{y - y_{\min}}{\text{GSD}} \right) \quad (1)$$

Where (x_{img}, y_{img}) are the pixel coordinates in the orthophoto, (x, y) are the geo-coordinates of a 3D point, (x_{min}, y_{min}) are the geo-coordinates of the bottom-left pixel on the orthophoto, and GSD is the Ground Sample Distance or Spatial resolution of the orthophoto.

Once each building’s point cluster is mapped to the orthophoto, a Convex Hull is applied to ensure that all the structural points belonging to a single building candidate are enclosed (Figure 7).

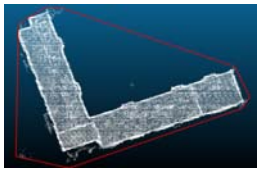


Figure 7. Building candidate points and corresponding image

These building candidates are further processed to ensure completeness and accuracy, serving as the foundational dataset for building-specific 3D modeling.

2.4 Roof Edge Detection

Roof edge detection is a critical step for delineating the break lines of each building. The Mobile LSD algorithm is employed to identify roof edges in each building candidate image. Identification of major line directions is carried out by applying line clustering techniques to ensure that dominant roof edges are accurately constructed. This step also eliminates or adjusts incorrect directions caused by noise or misidentifications during line detection, resulting in more reliable edge representations. The accuracy of this process is vital as it directly impacts the quality of the 3D reconstruction.

The Mobile LSD algorithm is chosen due to its robustness in detecting line segments in complex urban environments. Compared to traditional edge detection techniques such as Canny or Hough Transform; Mobile LSD offers higher robustness to noise and illumination variations, faster computation, and suitability for large-scale urban mapping. It also improves the detection of fine roof details, such as dormers and overhangs (Figure 8).

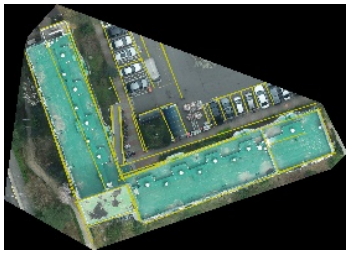


Figure 8. Roof edge detection using Mobile LSD

While Mobile LSD provides an initial set of line segments, raw detections may contain noise, missing edges, or misaligned segments. To improve detection accuracy, a clustering-based refinement process is implemented:

+ Clustering: Groups detected edges into clusters based on proximity and alignment.
+ Line Merging and Simplification: Parallel or closely spaced edges are merged into single continuous edges; Edge gaps due to occlusion or noise are interpolated.
+ Outlier Removal: Short, disconnected segments are filtered out based on a minimum edge length threshold.

2.5 Building Reconstruction

Roof edges derived from subsection 2.4 are extended and intersected each other to make bounding boxes with height attributes. Such attributes are assigned from the highest z-values of the corresponding points. By checking height information of the neighboring bounding boxes, these boxes are combined together to form 3D structures of buildings while ensuring that the reconstructed models reflect real-world structures.

The detected roof edges are extended into bounding boxes that define the spatial limits of the roof. This is implemented by:
+ Identifying roof lines from the Mobile LSD-based edge detection.
+ Expanding each roof segment to form a closed bounding box around the detected roof structure.
+ Dividing the bounding box into a grid. Each bounding box is subdivided into smaller grid cells. For each grid cell, the highest z-value from the non-ground point cloud within the cell is selected according to the equation (2).

$$z_{\text{cell},i} = \max(z_{\text{point cloud}} \in \text{cell}_i) \quad (2)$$

Once all grid cells in the bounding box have been assigned z-values, the final height of the bounding box is determined using a frequency-based approach. Specifically, the distribution of z-values across all grid cells within the bounding box is analyzed, and the most frequent z-value (mode) is selected as the representative height of the bounding box. After determining the z-value for each box, we then combine boxes with similar z-values to define a plane (Figure 9).

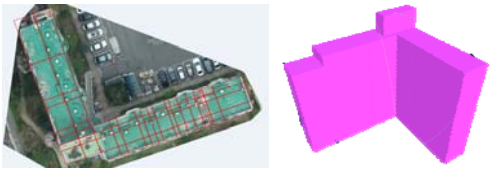


Figure 9. Bounding box and building model generation

3. Experimental Results

3.1 Pre-processing data

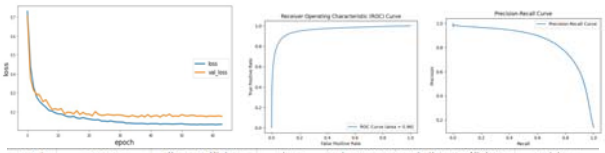
The area of interest is located in Suseo-dong, Gangnam-gu, South Korea. This area is characterized by numerous modern high-rise buildings and large-scale structures, including the Suseo Station, which plays a crucial role in transportation and urban development (Figure 10).

The data acquisition site is a 0.6 km radius area around Suseo Station in Suseo-dong, Gangnam-gu, Seoul, South Korea, covering approximately 1.13 km². A total of 1,328 UAV images were captured with a resolution of 5,472 × 3,648 pixels at an altitude of 130 meters. The flight was conducted with an image overlap and sidelap of 85% each to ensure high-quality photogrammetric processing. DJI Terra and PIX4D Mapper were used to generate the orthophotos and point clouds. The orthophoto has a GSD of 3.55 cm, while the point cloud has a density of 97,476 points per cubic meter.

Figure 10. Experiment area: Suseo area, South Korea

3.2 Building Candidate Generation

This study introduces enhancements to the U-Net model for improving tree segmentation from orthophotos. The modifications of the U-Net model discussed in subsection 2.2 lead to improvements in accuracy (Figure 11) and efficiency compared to the original U-Net model.



epoch	accuracy	dice_coefficient	loss	val_accuracy	val_dice_coefficient	val_loss
76	0.991965175	0.690122664	0.05541718	0.986890231	0.405686021	0.055025604
77	0.9925493	0.692189515	0.054638613	0.989357173	0.419843584	0.052812509
78	0.99266535	0.694415569	0.054470543	0.989261866	0.423004538	0.052559812
79	0.993065953	0.696440458	0.053829532	0.989074707	0.424370199	0.052172482
80	0.991633713	0.691772223	0.055496279	0.989161849	0.421057016	0.051935386
81	0.991925359	0.693962514	0.055018015	0.989144921	0.405663937	0.056617703
82	0.992993414	0.699737608	0.053468433	0.989338517	0.418288261	0.054867748
83	0.993599296	0.701935232	0.052589305	0.989323258	0.414741665	0.056582335
84	0.993882179	0.703439593	0.052097224	0.98923105	0.428594261	0.054011706
85	0.994081676	0.707074463	0.051732998	0.989234746	0.427058488	0.055509858
86	0.994253218	0.708123565	0.051376466	0.989399552	0.420545608	0.058307268
87	0.992708743	0.703199691	0.053171411	0.986568868	0.386079103	0.062252924
88	0.978939891	0.640262246	0.07372608	0.982098043	0.229686603	0.050641134
89	0.97163564	0.578445613	0.088233896	0.989434838	0.427545696	0.032040611
90	0.986165047	0.664883494	0.062207218	0.988991976	0.421421558	0.040402539
91	0.990591586	0.691288054	0.055677816	0.988704741	0.392903835	0.051905617
92	0.992419422	0.702579319	0.053125851	0.989166617	0.414514095	0.052425183
93	0.993590593	0.707154393	0.05151286	0.989330351	0.418654203	0.05420962
94	0.993556976	0.709697664	0.051277805	0.989476502	0.428640038	0.053851556
95	0.994083762	0.713078976	0.050550304	0.989220202	0.422329456	0.056135442
96	0.99424696	0.714238644	0.050252538	0.988735259	0.425907582	0.054975674
97	0.994547904	0.716278255	0.049758114	0.989228666	0.410474807	0.060924806
98	0.994804204	0.719522417	0.049313519	0.989225268	0.424901277	0.058081035
99	0.995093107	0.720633507	0.048840228	0.989128649	0.432987332	0.056902334

Figure 11 . Enhanced U-Net model accuracy

The enhanced U-Net model, which integrates layer normalization after each convolution, demonstrates stable training and strong performance. The loss and validation loss curves show steady improvement over epochs, indicating effective convergence. Both the ROC and Precision-Recall curves suggest high predictive accuracy, with an AUC nearing 1.0. Additionally, the table in Figure 11 highlights the model’s reliability by presenting consistently high accuracy, Dice coefficient, and validation scores during the final epochs. These results confirm the model’s robustness in segmentation tasks and the effectiveness of the architectural enhancements. Figure 12 shows the tree segmentation results using the enhanced U-Net model.

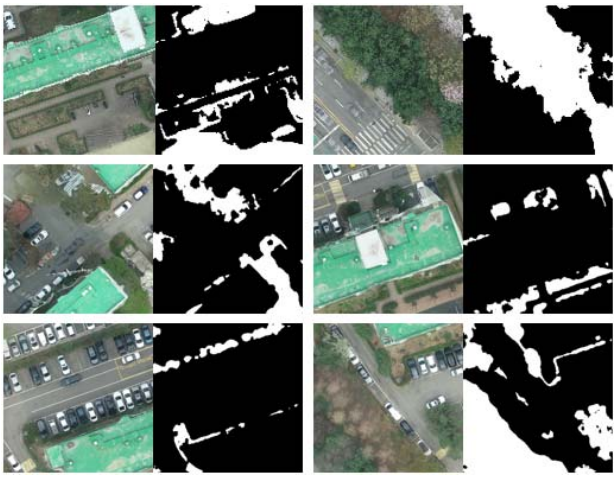


Figure 12. Tree segmentation results

Figure 13 shows the separated ground and non-ground point clouds from the original point clouds. Also, Figure 14 shows the outcomes after applying three removal and height-filter on the non-ground point clouds.

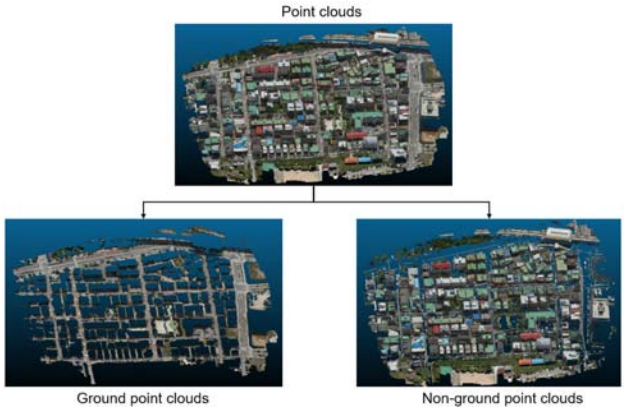


Figure 13. Point clouds separation using CSF algorithm

Figure 14. Before and after applying tree removal and height-filter

By applying tree segmentation, CSF, and height-based filtering algorithms, non-building objects such as ground, trees, cars, humans, and other small objects are filtered out while leaving only significant structures like buildings.

Afterwards, a clustering technique is carried out to separate the building point cloud into individual clusters, each representing a single building candidate (Figure 15).

Figure 15. Building candidate extraction using DBSCAN clustering algorithm

Afterward, the Convex Hull algorithm is applied to each building candidate, and the corresponding building image is extracted from the orthophoto. Figure 16 shows several results of building image candidates in Suseo area.

Figure 16 .Building image candidates

3.3 3D Building Model Reconstruction

The Mobile LSD algorithm is employed to detect roof edges from the building images. Line clustering is then applied to identify the major line directions from the line segments, ensuring that dominant roof edges are accurately captured. This step helps refine the results by eliminating or correcting misaligned directions caused by noise, producing clean and precise edge lines. These refined lines serve as the foundation for constructing bounding boxes and modeling building structures.

Using the detected roof edges and bounding boxes, 3D building models are reconstructed. Height attributes for each box are determined using point coverage calculated through the alpha shape algorithm. This method accurately identifies z-values by analyzing the spatial distribution of points within each box, enabling the generation of detailed 3D planes for each building.

The final models accurately represent the spatial and structural characteristics of the buildings, showcasing the success of the integrated workflow (Figure 17).

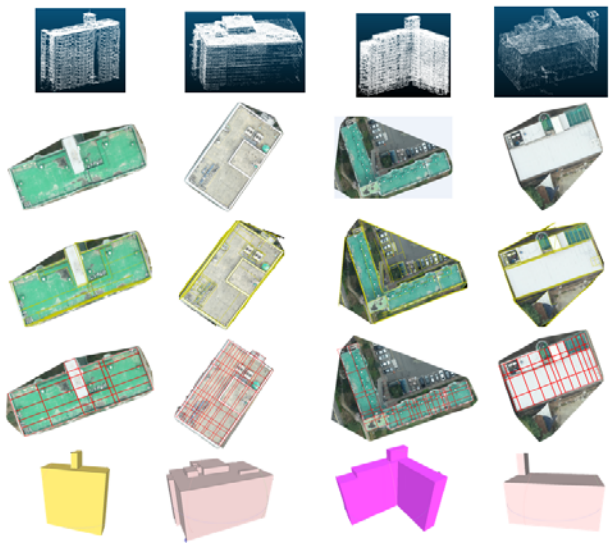


Figure 17. Examples of reconstructed building models in Suseo area

The quantitative measures of the 3D building reconstruction in this context are computed using the following equations:

$$\text{Precision} = \frac{\text{Number of Correct Planes}}{\text{Our Method's Number of Planes}} \quad (3)$$

$$\text{Recall} = \frac{\text{Number of Correct Planes}}{\text{Ground truth}} \quad (4)$$

$$\text{F1} = \frac{\text{Precision x Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Where Precision represents the proportion of correctly predicted positive instances out of all instances predicted as positive. Recall represents the proportion of correctly predicted positive instances out of all actual positive instances. F1 represents the harmonic mean of Precision and Recall. Table 1 shows the quantitative evaluations carried out using the four buildings shown in Figure 17.

Table 1 summarizes the accuracy of plane detection in 3D building reconstruction, comparing the proposed method to ground truth across four buildings.

	Number of plane		Number of correct plane	Precision	Recall	F1
	Ground truth	Our method				
Building 1	10	10	10	1.0	1.0	1.0
Building 2	24	32	20	0.625	0.833	0.714
Building 3	13	13	13	1.0	1.0	1.0
Building 4	16	10	10	1.0	0.625	0.769

Table 1 . Quantitative Analysis for Planes in 3D reconstruction

For simpler structures (Buildings 1 and 3), the method achieves perfect results, with precision, recall, and F1 scores of 1.0. In more complex cases (Buildings 2 and 4), either recall or precision is slightly reduced. For Building 2, while the proposed method captures most ground truth planes, it also identifies some additional ones. In the case of Building 4, which has many structures on the roof, the method misses some planes. This demonstrates the method's robustness while also highlighting challenges in handling complex structures.

4. Conclusions

This study demonstrates an effective workflow for 3D building reconstruction using UAV-derived orthophotos and point clouds. By integrating deep learning, clustering, and geometric approaches, the research ensures accurate building candidate extraction, roof edge detection, and 3D building modeling.

The results highlight the synergy of integrating orthophotos and point clouds, combining contextual details with geometric precision. Future work will focus on advancing the current outcomes, toward more detailed and complex models up to the Level of Detail 3. This involves integrating more refined architectural details and tackling challenges in modeling complex building structures to improve applicability in dense urban environments.

Acknowledgment

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant R&D program of Digital Land Information Technology Development funded by the Ministry of Land, Infrastructure and Transport (MOLIT) (Grant RS-2022-00142501).

References

Chan, J., and Qin, R. (2017). "Tree extraction methods for UAV imagery." doi.org/10.1017/tree_extraction_methods (10 Feb 2025)

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise." In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD), pp. 226–231.

Gómez, J., and Téllez, F. (2020). "UAV-based photogrammetry for urban applications: Challenges and solutions." Journal of Photogrammetric Science, 48(3), 205–215.

Gómez, R., and Tellez, J. (2020). "Orthophotos and point clouds: An integrated approach to urban reconstruction." In: ISPRS Annals of Photogrammetry, Remote Sensing, and Spatial Information Sciences, V-2-2020, 25–30.

Lawlor, J., and O'Keeffe, E. (2018). "A Review of UAV Photogrammetry for Archaeology and Building Documentation." Remote Sensing, 10(11), 1755.

Michalis, P., and Dowman, I. (2008). "Integration of photogrammetric and LiDAR data for urban modeling." Photogramm. Rec., 23(121), 20–30.

Preparata, F. P., and Hong, S. J. (1977). "Convex Hulls of Finite Sets of Points in Two and Three Dimensions." Communications of the ACM, 20(2), 87–93.

Barber, C. B., Dobkin, D. P., and Huhdanpaa, H. (1996). "The Quickhull Algorithm for Convex Hulls." ACM Transactions on Mathematical Software (TOMS), 22(4), 469–483.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation." In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234–241. Springer, Cham.

Schubert, E., Sander, J., Ester, M., Kriegel, H.-P., and Xu, X. (2017). "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN." ACM Transactions on Database Systems (TODS), 42(3), 19:1–19:21.

Shapira, L., Aiger, D., and Cohen-Or, D. (2008). "Consistent Mesh Partitioning and Skeletonization Using the Shape Diameter Function." Visual Computer, 24(4), 249–259.

Edelsbrunner, H., Kirkpatrick, D. G., and Seidel, R. (1983). "On the Shape of a Set of Points in the Plane." IEEE Transactions on Information Theory, 29(4), 551–559.

Wang, H., and Liu, Z. (2023). "Advances in UAV-based geospatial data processing: Opportunities and challenges." GIScience Journal, 59(1), 15–32.

Zhang, L., and He, Y. (2022). "High-resolution 3D building reconstruction using UAV photogrammetry and LiDAR." ISPRS J. Photogramm. Remote Sens., 187, 14–27.

Zhang, Y., Smith, J., and Roberts, P. (2022). "Applications of UAV imagery in urban analysis: A review." Urban Data Journal, 10(4), 540–556.

Zhao, Z., Qin, X., and Lin, Y. (2016). "Cloth Simulation Filter for point cloud ground segmentation." International Journal of Remote Sensing, 37(5), 1184–1202.

Zhou, F., Zhou, J., and Cao, X. (2021). "MobileLSD: Lightweight Line Segment Detection for Resource-Constrained Environments." arXiv preprint arXiv:2106.00186.

Zhou, F., He, Q., and Wu, X. (2022). "Evaluation of MobileLSD for 3D line segment detection in UAV data." Remote Sensing Letters, 13(7), 651–660.