



You Only Look Once Version 11 (YOLOv11) Based Object Detection for 3D City Modeling: A Study in the Jatirejo Area

Ni Putu Atmelia Putri, Monica Maharani*

Universitas Pembangunan Nasional Veteran Yogyakarta, Indonesia

Received : September 15,
2025

Revised : October 1,
2025

Accepted : October 3, 2025

Online : October 14, 2025

Abstract

Digital three-dimensional (3D) modeling has become an essential requirement in modern spatial mapping and visualization, as it can provide a more realistic and detailed representation of objects or areas. This study uses image segmentation techniques using YOLO v11, which automatically recognizes and separates objects, thereby increasing the accuracy of image extraction and accelerating the modeling process. With the help of software and the web, namely Roboflow, Google Colab, and QGIS. The results of this study show that the integration of image extraction algorithms, deep learning, and image segmentation based on YOLO v11 produces a more precise, efficient, and realistic 3D model. The Confusion Matrix shows that the segmentation results are perfectly detected at a rate of more than 85.64%, with the remaining segmentation not being perfectly detected, accounting for 9.23% and 5.13% of the undetected part. The calculation of the precision value of 95.94% indicates that the model rarely makes mistakes in predicting objects. The resulting Recall value is 87.58% and the F1 score is 91.57%. Thus, the use of AI-based technology and computer vision offers an innovative solution in accelerating the development of effective, accurate, and cost-effective digital 3D city models that can be used as local government data.

Keywords *Segmentation, Yolov11, 3D City, Extraction*

INTRODUCTION

In recent years, 3D modeling has provided a more realistic representation or visualization compared to two-dimensional modeling, making it easier for people to understand the shape, structure, and other information of the modeled object. 3D modeling has become a necessity in various fields, including research, mapping, visualization, inventory management, and maintenance (Priambodo et al., 2022). One area that utilizes 3D modeling is 3D urban modeling. These 3D models offer varying levels of detail and represent the shape of buildings within a city (Singh et al., 2013). The method used in this modeling is photogrammetry. Photogrammetry can obtain information about the position, size, and shape of an object without requiring direct measurement. Close-range photogrammetry techniques were employed to represent objects with a height of less than 100 meters, with the camera positioned in close proximity to the object being represented (Surahman et al., 2021).

3D modeling in photogrammetry can be performed using UAVs and specialized software to process the resulting data. Data collection is also relatively fast, as a single drone flight can produce hundreds of photos, suitable for both urban and rural areas. With software support, data processing in 3D modeling can also be completed in a relatively short time. Therefore, photogrammetry is one of the most practical, efficient, and effective methods for producing 3D models with a high degree of accuracy and informative visualizations (Priambodo et al., 2022).

One way to achieve automatic segmentation is by training data using the deep learning-based object detection algorithm, YOLOv11 (You Only Look Once). YOLOv11 adheres to the

Copyright Holder:

© Ni Putu & Monica. (2025)

Corresponding author's email: monica.maharani@upnyk.ac.id

This Article is Licensed Under:



principle of real-time detection, enabling the detection of objects within milliseconds. This is a key advantage of YOLOv11, which boasts an extremely high processing speed (Hendriko & Hermanto, 2025). YOLOv11 also features improvements to its backbone architecture, resulting in improved accuracy, more flexible anchor-free use, and the integration of FPN (Feature Pyramid Network) and PAN (Path Aggregation Network) to create multi-scale features. This combination enables YOLOv11 to detect both large and small objects with high precision (Kıratlı & Eroğlu, 2025). In addition to the ability to detect objects with bounding boxes on free anchors, YOLOv11 features an instant segmentation capability that generates pixel-precise masks, following the shape of the object (Sapkota & Karkee, 2025). This data training process is assisted by Roboflow and Google Colab, which facilitate easy data access and processing.

In this study, the Jatirejo, Sendangadi, Mlati, Sleman area was used as a 3D city modeling area using the YOLOv11 algorithm. The Jatirejo area itself has a relatively high population density. Houses are quite close together. This area also features many housing complexes and lodgings, resulting in very close distances between buildings. Therefore, this area was chosen as sample data or training data for algorithm development. The YOLOv11 rhythm in segmenting adjacent buildings was also tested. It also tested the ability of YOLOv11 to detect house objects in dense areas with irregular distribution. The purpose of this study is to evaluate and apply YOLOv11-based segmentation to enhance the accuracy of building detection in densely populated settlements, thereby supporting the development of more reliable 3D city models for urban planning, infrastructure management, and sustainable spatial development.

LITERATURE REVIEW

3D reconstruction uses various methods, including classical photogrammetry, Structure from Motion (SfM), Multi-View Stereo (MVS), and deep learning-based methods. The photogrammetric modeling process begins with the capture of aerial photographs using an aircraft equipped with a calibrated camera. The aerial photographs are arranged in a mosaic with a certain degree of overlap, allowing the formation of stereo pairs that enable objects to be viewed in a three-dimensional model using a stereoscope (Arif et al., 2025; Prasetyo, 2018). The structure of the movement (SfM) and Multi-View Stereo (MVS) are closely related in 3D reconstruction from images/photos (Grohmann et al., 2023). SfM can calculate camera orientation (position and direction of shooting) while simultaneously building the initial structure of an object in the form of a Sparse Point Cloud (still far away). SfM produces a point cloud that is still far away, then MVS is used to enrich the results into a dense point cloud. MVS works by utilizing information from many photos that have high overlap. Each pixel in a photo is matched with other pixels in different photos to estimate its depth. This process produces hundreds of thousands to millions of 3D points that form the surface details of the object (Pepe et al., 2022).

The development of deep learning has brought significant changes to the field of 3D reconstruction. While classic photogrammetry, SfM, or MVS methods rely on the principles of projection geometry and feature matching, deep learning relies more on artificial neural networks. Deep learning is a learning method that utilizes several nonlinear transformations. Deep learning can be thought of as a combination of machine learning and AI (artificial neural networks) (Chauhan & Singh, 2018). This model operates by training on a dataset comprising pairs of 2D images and their corresponding 3D representations. This model learns to infer missing depth information from photographs, enabling it to estimate 3D shapes from a single image. Examples of deep learning methods are NeRF (Neural Radiance Fields) and YOLO. NeRF (Neural Radiance Fields) exploits the concept of light in photographs coming from multiple directions in three-dimensional space, whereas NeRF works implicitly with volume rendering, resulting in highly realistic images that can be rendered from any perspective. YOLO is a machine learning algorithm known for its ability to

accurately detect objects or faces (Maharani et al., 2025; Wang et al., 2023). YOLO uses an artificial neural network approach that can detect objects in images or photos (Santos et al., 2022). This network divides the image into several regions and detects objects in each region.

YOLO divides the input image into a grid, with each cell predicting the presence of an object. If an object is present, the payer grid will predict its class (Jiang et al., 2021). YOLOv11 uses a backbone network to extract features from images. The extracted features include edges, textures, roof patterns, wall shapes, and building characteristics. YoloV11 utilizes the Pyramid Network (FPN) and Path Aggregation Network (PAN) features to detect objects across a range of scales, from small to large. Sometimes, overlapping bounding boxes detect the same building. Hence, NMS (Non-Maximum Suppression) selects the box with the highest score and removes other similar boxes to achieve more accurate detection results (Zhang, 2023). YOLO will automatically generate building segmentation based on the image/photo data used. To visualize it as a 3D model, you can use Qgis2threejs. By adding the height of each building, you will visualize the buildings that YOLO has segmented.

```
!pip install --upgrade ultralytics
!pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="Rt2RTslGyRrXjnMV023C")
project = rf.workspace("puput-m7tfq").project("segmentasi-paper-jgiaz")
version = project.version(6)
dataset = version.download("yolov11")

from ultralytics import YOLO

# Load a model
model = YOLO('yolo11n.pt') # load a pretrained model (recommended for training)

train_results = model.train(
    data="/content/SEGMENTASI-PAPER-6/data.yaml",
    epochs=100,
    imgsz=640,
    batch=4,
    patience=0,
    mask_ratio=1,
    cls=1.0,
    device=0,
)
```

Figure 1. YOLOv11 Algorithm

RESEARCH METHOD

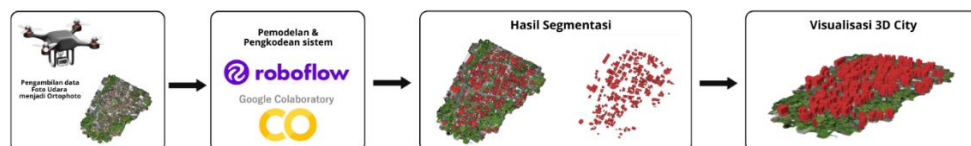


Figure 2. Flowchart

Concept Planning

The research location was Jatirejo Hamlet in Sendangadi, Mlati, Sleman, which has a relatively dense housing density. Primary data were obtained through aerial photography using a Mavic Air 2S drone equipped with a high-resolution camera. As shown in Figure 3, the drone was flown along a trajectory with 80% grid overlap and 70% sideslip. The flight altitude was 90 meters above the surface, reflecting the characteristics of the Sendangadi lowland location. The drone flew

from 11:00 to 1:00 PM WIB. This flight trajectory produced 167 photographs.



Figure 3. Drone Deploy Flight Path

System Modeling

Aerial photos processed into orthomosaics in Agisoft with medium to low specifications will produce orthophotos that are not too sharp, allowing them to be easily identified by the model, and with minimal differences in color and texture. The deep learning-based image extraction process with YOLOv11 begins by uploading orthophotos to the Roboflow platform. The labeling process is carried out on building objects, which can use AI assistance (automatic) or be done manually on the building formations found in Roboflow tools. Using the Dataset Method, namely Split Image Between Train/Validation. By setting the Dataset Split to 70% Training and 30% Validation, because the training data is only 25 images, the Test section is removed, and the validation section is increased to be able to validate a large amount of data.

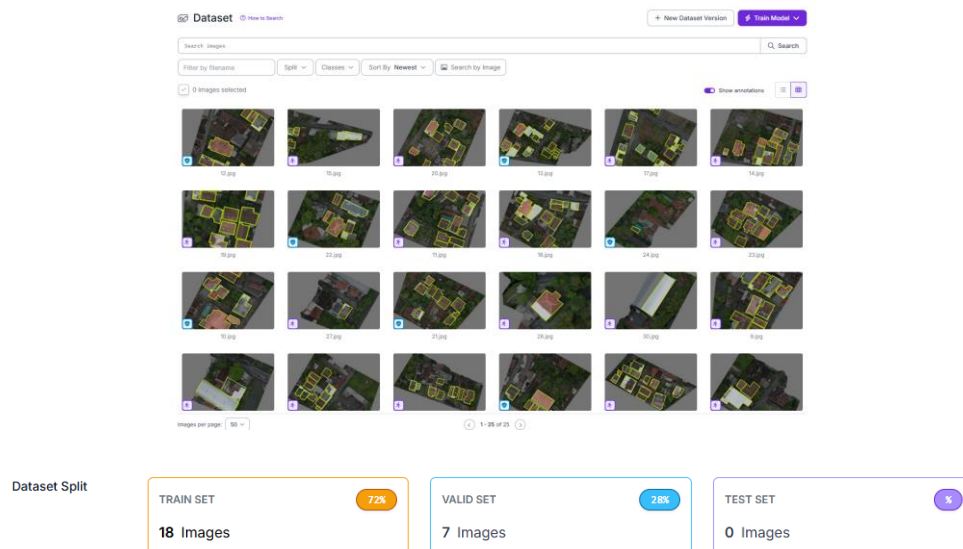


Figure 4. Annotation and Split Data

Encoding

After the dataset or training data was processed in Roboflow, the next step was to connect it to Google Colab. Using the Python programming language, we imported the dataset prepared by Roboflow into Google Colab through the provided API. This dataset was adapted to the YOLO V11 format, which has improved the speed and accuracy of object detection compared to other algorithms. A total of 100 epochs were run because the results were very satisfactory (Ali & Zhang, 2024).

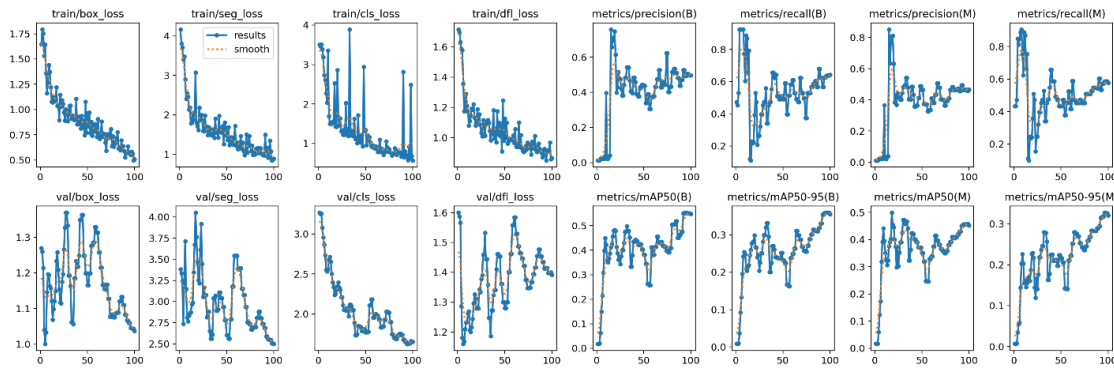



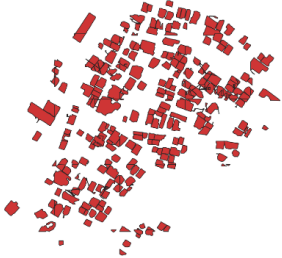
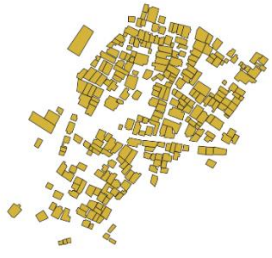
Figure 5. Segmentation Results Graph

The graph above shows a continuous downward trend at the beginning, indicating good performance in segmentation, box prediction, object classification, and bounding box accuracy. On the loss function side, the box_loss, cls_loss, and df_loss graphs consistently decrease, indicating that the model has successfully learned important patterns in the dataset. The mAP50 for bounding boxes is higher than for masks, while mAP50-95 is lower for both. Overall, the model demonstrates good learning performance with a small dataset, as evidenced by the decrease in loss and increase in mAP. The results were converted to GeoJSON for easy processing in QGIS for 3D model visualization.

FINDINGS AND DISCUSSION

The segmentation generated using the pre-trained model from RoboFlow performed quite well. The results showed that most buildings were successfully detected according to their original shapes in the provided aerial images, but some buildings were not perfectly segmented. This imperfection in building segmentation could be caused by the segmentation method being performed incrementally on each image segment, rather than on the entire aerial image. Furthermore, this could also be due to weaknesses in RoboFlow's data training process, which lacked sufficient training data to inform the model. As a result, some buildings were not properly extracted.

Table 1. Segmentation Results in QGIS

Orthophoto	Building Segmentation	Ground Truth
		

Building Footprints are Well-Detected**Figure 6.** Example of Well-Detected Building Locations (A) Segmentation, (B) Orthophoto, (C) Ground Truth

Figure 5 illustrates the correspondence between the buildings identified in the segmentation results and the conditions depicted in the aerial photographs, as shown in Figure 2. These results indicate that the segmentation algorithm used has accurately identified the main characteristics of the building, such as its roof shape, edges, and texture.

Detection Error

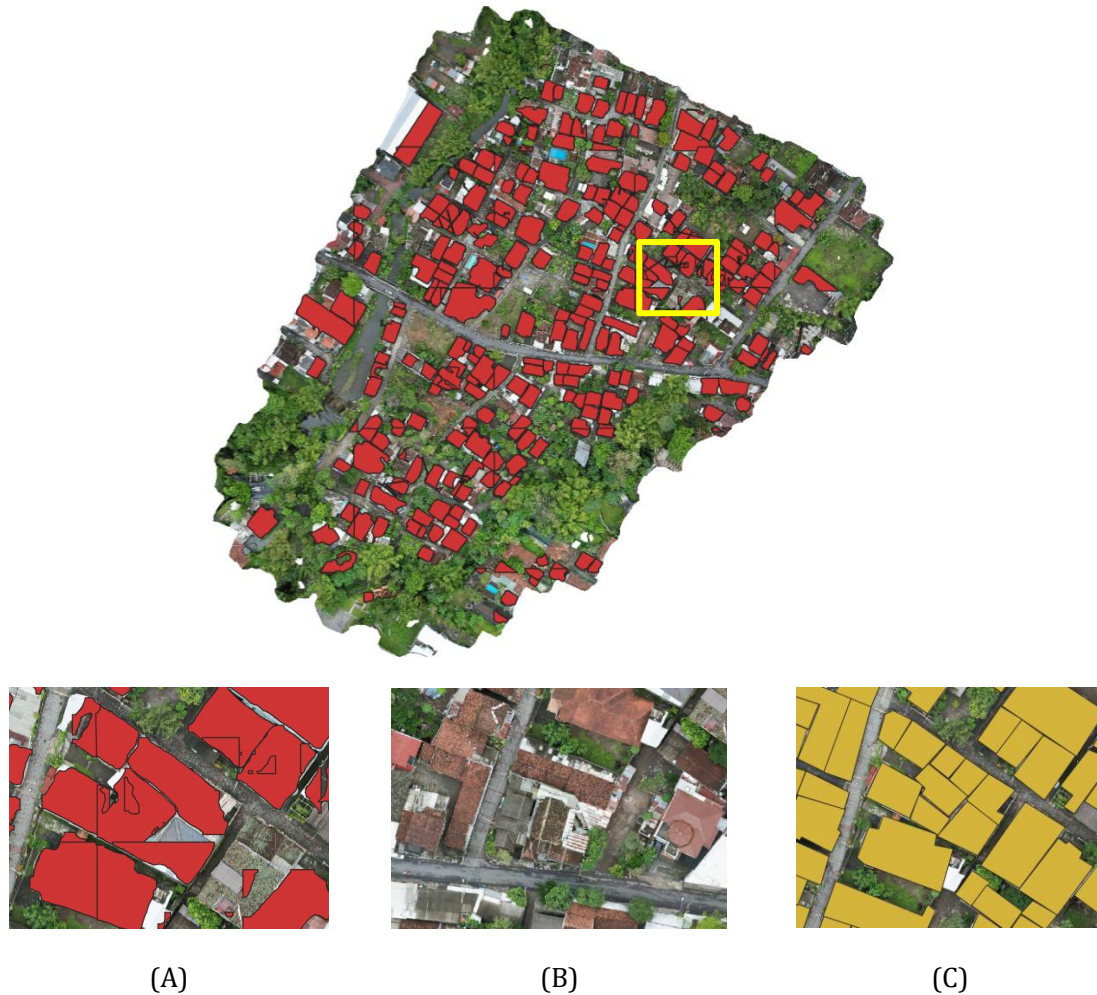


Figure 7. Example of Well Detected Building Location (A) Segmentation, (B) Orthophoto, (C) Ground Truth

An error occurred, indicating that the segmentation was truncated or not fully detected, encompassing the entire building. This was caused by the segmentation process being performed at an intersection, resulting in some of the building's footprint being overlooked.

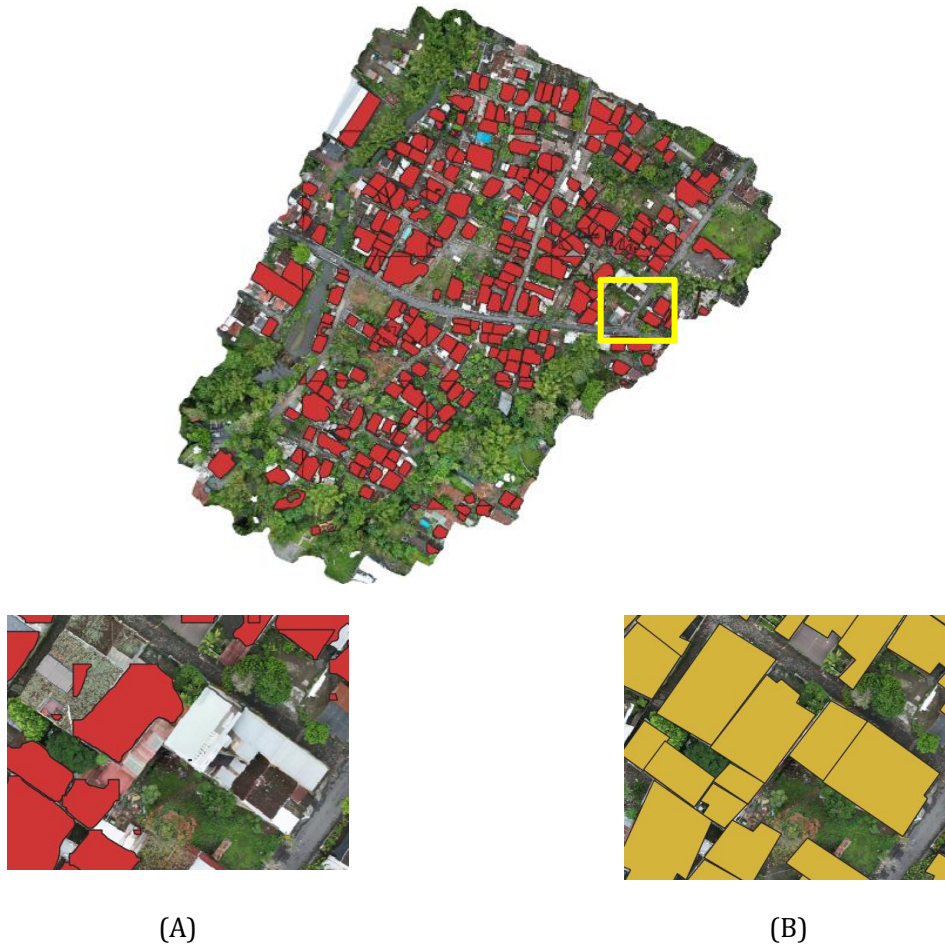
Not Detected

Figure 8. Example of Undetected Building Traces (A) Segmentation, (B) Ground Truth

In the image above, several buildings remain that the model did not detect. This is due to the segmentation model's inability to recognize the color and shape characteristics of certain areas.

3D Visualization

Figure 9. 3D City Visualization

3D City Visualization is achieved with the help of QGIS software, utilizing the Qgis2threejs plugin. This plugin can leverage the OSM Building website as a source of surface elevation data, which is then combined with vector or raster layers to produce a 3D model that accurately represents actual field conditions. Satellite imagery or orthophotos can be used as textures on the DEM surface, making the visualization look more realistic. The visualization results are displayed in a new web-based window that allows users to rotate, zoom, or move the camera to create a more realistic view. This is one of the advantages of Qgis2threejs.

Accuracy Test

The confusion matrix provides an overview of the distribution of errors and the correctness of the model's classification. User accuracy specifically assesses the user's level of confidence in the prediction results of a class, allowing users to see not only the general accuracy but also the quality of detection in each class.

$$\text{User's Accuracy} = x \times 100\% \frac{xii}{x+i}$$

Table 2. Confusion Matrix

CONFUSION MATRIX						
Class	FIELD TESTING			Total Field Samples	Producer's Accuracy	
	Perfectly Detected	Completely Undetectable	Completely Undetectable			
CSRT CLASSIFICATION	Perfectly Detected	167	167	167	195	85.64102564
	Completely Undetectable	18	18	18	195	9.230769231
	Completely Undetectable	10	10	10	195	5.128205128
	Total sample classification	195	195	195		
	user's accuracy	85.64102564	9.230769231	5.128205128		

Table 2 shows that using YOLOv11 training data and processing with Python can produce high-quality automatic segmentation of orthophotography data. Of the 195 buildings manually digitized, 167 were perfectly detected with this model. Another 18 buildings were also detected, although not perfectly, due to variations in color on the roofs. Finally, 10 buildings were not detected because they were obscured by trees, or the building did not detect their color and texture. Based on the polygon segmentation attribute table data, 287 polygons were recorded, with approximately 92 of them being anomalies, such as trees, fields, or other objects outside the building that share the same texture, color, and shape as the model. These anomalies can also be overlay polygons within the building.

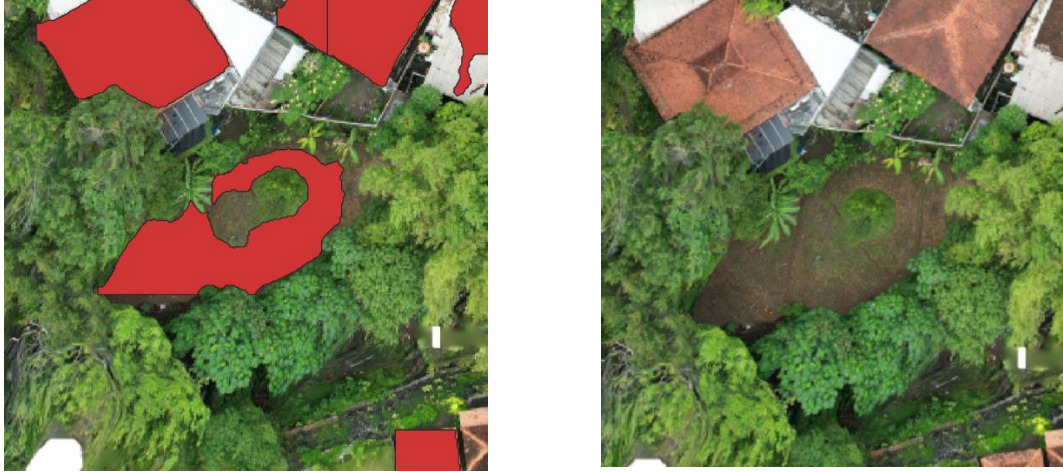


Figure 10. Non-Building Anomalies

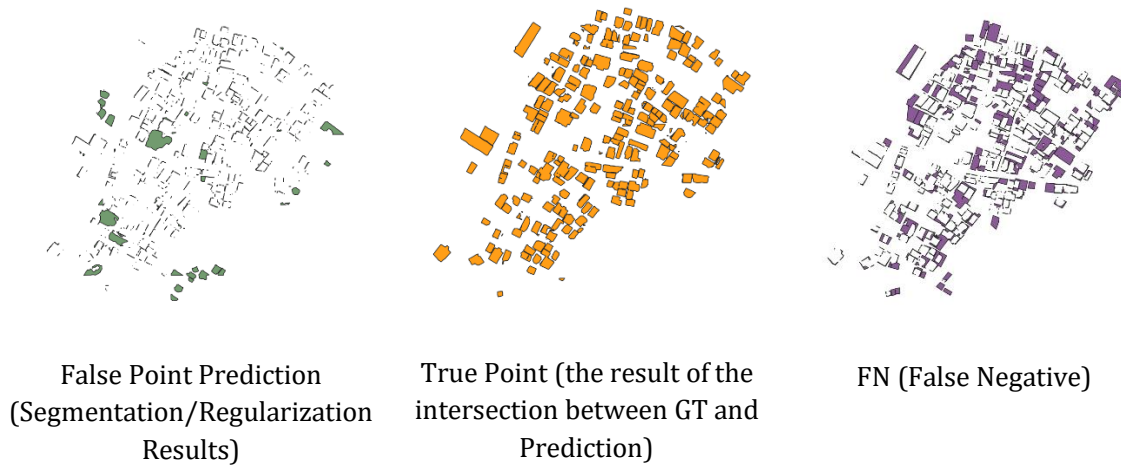


Figure 11. Overlay Anomaly

Table 3. Model Performance Evaluation

	Segmentation Results	
	S1	
TP (m²)	46397.586	
FP (m²)	1961.243	
FN (m²)	6581.669	
Precision	95.94%	
Recall	87.58%	
F1-Score	91.57%	

Table 3 shows a precision value of 95.94%, indicating that the model rarely predicts the wrong object. However, the low recall value of 87.58% indicates that there are still areas of the object that are not detected. Overall, the model's F1 score of 91.57% demonstrates a strong balance between accuracy and completeness of detection.

Table 4. Visualization of Model Performance Evaluation

CONCLUSIONS

Automatic segmentation using the YOLOv11 deep learning method, assisted by Roboflow and Google Colab, with an image extraction approach, demonstrates that the model achieves a high detection performance, with mAP50 on the bounding box being higher than the mask, while mAP50-95 is lower for both. Overall, the model shows good learning ability with a small dataset as indicated by a decrease in loss and an increase in mAP. This results in an accuracy rate of 85.64% for perfectly detected segmentations, specifically 167 buildings accurately detected.

This achievement indicates that this automatic segmentation method is effective in identifying objects in most of the test data. However, there are still 9.23% imperfect segmentation results, namely 18 buildings, and 5.13% that failed to be detected, namely 10 buildings. The results of the model performance evaluation show a precision value of 95.94%, indicating that most of the areas predicted by the model match the target object. However, the recall of 87.58% suggests that the model's ability to find all real objects is still not perfect. Therefore, the success of this segmentation model depends on the amount and quality of the training data used, the choice of syntax for processing the data using Python, and the quality of the automatically segmented data. This model also tends to be "cautious" in its predictions, resulting in fewer false detections in non-object areas. YOLOv11 can be a tool for effective object detection and accurate 3D city modeling.

This research, similar to [He et al. \(2025\)](#), also used the YOLOv11 model to train, detect, and identify land cover targets in remote sensing imagery. After 496 training epochs, the metrics achieved a precision of 0.8861, a recall of 0.8563, a map50 of 0.8920, a map50-95 of 0.8646, and an F1 score of 0.8709, indicating robust and consistent performance. The model in this study is less suitable for large-scale implementation, such as that carried out by [He et al. \(2025\)](#), but this model is suitable for creating a 3D City in areas that are not too large and will get more detailed results and can be used in sub-district/urban village areas to obtain land registration data for each resident in that area.

LIMITATIONS & FURTHER RESEARCH

In future research, data from various sources, such as LiDAR, high-spec drones, or 3D laser scanning data, can be utilized to enhance the detail and accuracy of digital city models, enabling real-time segmentation and modeling to support practical applications, including traffic monitoring, large-scale smart cities, and disaster mitigation. This study has several limitations. First, this study focused solely on building objects that can be recognized through image segmentation techniques, rather than covering all elements of city detail (e.g., road surface texture,

trees, and other small details). Second, the data is limited to two-dimensional (2D) imagery extracted from the Mavic Air 2 drone source. There was no testing with Lidar data or large-scale aerial photogrammetry, and the specifications of the drone used limit the coverage area. Finally, the resulting 3D model is representative for visualization purposes and basic spatial analysis. It has not been tested for detailed engineering applications that require a high level of precision. This model can serve as a reference for further research in 3D mapping and image segmentation. Its primary implication is the use of sub-district or village-scale data to record land ownership within a given area.

REFERENCES

- Ali, M. L., & Zhang, Z. (2024). The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection. *Computers*, *13*(12), Article 336. <https://doi.org/10.3390/computers13120336>
- Arif, R.S., Harintaka, & Maharani, M. (2025). Investigating the impact of enhanced images on 3D reconstruction of non-Lambertian objects using neural radiance fields. *IOP Conference Series: Earth and Environmental Science*, *1486*(1), 012024. <https://doi.org/10.1088/1755-1315/1486/1/012024>
- Chauhan, N. K., & Singh, K. (2018). A review on conventional machine learning vs deep learning. *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, 347–352. <https://doi.org/10.1109/GUCON.2018.8675097>
- Grohmann, C. H., Viana, C. D., Garcia, G. P. B., & Albuquerque, R. W. (2023). Remotely piloted aircraft-based automated vertical surface survey. *MethodsX*, *10*, 101982. <https://doi.org/10.1016/j.mex.2022.101982>
- He, L. H., Zhou, Y. Z., Liu, L., Cao, W., & Ma, J. H. (2025). Research on object detection and recognition in remote sensing images based on YOLOv11. *Scientific Reports*, *15*(1), Article 96314. <https://doi.org/10.1038/s41598-025-96314-x>
- Hendriko, V., & Hermanto, D. (2025). Performance comparison of YOLOv10, YOLOv11, and YOLOv12 models on human detection datasets. *Brilliance: Research in Artificial Intelligence*, *5*(1), 440–450. <https://doi.org/10.47709/brilliance.v5i1.6447>
- Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2021). A review of YOLO algorithm developments. *Procedia Computer Science*, *199*, 1066–1073. <https://doi.org/10.1016/j.procs.2022.01.135>
- Kıratlı, R., & Eroğlu, A. (2025). Real-time multi-object detection and tracking in UAV systems: Improved YOLOv11-EFAC and optimized tracking algorithms. *Journal of Real-Time Image Processing*, *22*(5), 178. <https://doi.org/10.1007/s11554-025-01758-z>
- Maharani, M., Arif, R., Pradana, A., Ayudhia, K., Afifah, L., & Ikram, N. (2025). Comparative analysis of 3D reconstruction results using MVS and NeRF: Insights from PCA. *IOP Conference Series: Earth and Environmental Science*, *1486*(1), 012020. <https://doi.org/10.1088/1755-1315/1486/1/012020>
- Pepe, M., Alfio, V. S., & Costantino, D. (2022). UAV platforms and the SfM-MVS approach in the 3D surveys and modelling: A review in the cultural heritage field. *Applied Sciences (Switzerland)*, *12*(24), 12886. <https://doi.org/10.3390/app122412886>
- Prasetyo, Y. (2018). *State-of-art conservation of buildings and cultural heritage through 3-dimensional model formation based on near-range photogrammetric techniques*. <https://doi.org/10.14710/elipsoida.2018.3698>
- Priambodo, D. A., Tjahjadi, M. E., & Suhari, K. T. (2022). Making a 3D model of the Bayat Highway for existing purposes using the aerial photography (UAV) method in Klaten. *Teras Jurnal: Jurnal Teknik Sipil*, *12*(1), 177–190. <https://doi.org/10.29103/tj.v12i1.654>
- Santos, C., Aguiar, M., Welfer, D., & Belloni, B. (2022). A new approach for detecting fundus lesions

- using image processing and deep neural network architecture based on YOLO model. *Sensors*, 22(17), 6441. <https://doi.org/10.3390/s22176441>
- Sapkota, R., & Karkee, M. (2025). Comparing YOLOv11 and YOLOv8 for instance segmentation of occluded and non-occluded immature green fruits in complex orchard environment. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2410.19869>
- Singh, S. P., Jain, K., & Mandla, V. R. (2013). Virtual 3D city modeling: Techniques and applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(2/W2), 73–91. <https://doi.org/10.5194/isprsarchives-XL-2-W2-73-2013>
- Surahman, A., Wahyudi, A. D., Putra, A. D., Sintaro, S., & Pangestu, I. (2021). *InfoTekJar: National Journal of Informatics and Network Technology*, 5(2). <https://doi.org/10.30743/infotekjar.v5i2.3305>
- Wang, X., Yuan, L., Sun, L., Wu, S., & Liu, A. (2023). Three-dimensional object segmentation method based on YOLO, SAM, and NeRF. *Proceedings of the 2023 International Conference on Computer, Vision and Intelligent Technology*. <https://doi.org/10.1145/3627341.3630370>
- Zhang, Z. (2023). Drone-YOLO: An efficient neural network method for target detection in drone images. *Drones*, 7(8), 526. <https://doi.org/10.3390/drones7080526>