A data-driven approach to add openings to 3D BAG LOD2 building models

Yitong Xia
student #5445825

1st supervisor: Dr. Jantien Stoter
2nd supervisor: Weixiao Gao
External supervisor: Revi Peters

January, 2023

Contents

1 Introduction

This chapter presents an overview of 3D city models, including their definitions, advantages, and primary application scenarios. It also reviews the mainstream methods for generating 3D city models and how researchers utilize the concept of level of detail (LOD) to represent the varying degrees of detail inherent in 3D city models. Additionally, this chapter discusses the prevalent approaches for extracting façades and openings from imagery.

In research related to the urban environment, there is an increasing demand for accurate, comprehensive, and up-to-date representations of buildings utilizing 3D models. A 3D city model is a digital representation and simulation of the urban environment using three-dimensional geometry [Batty et al. (2001); Peters et al. (2022); Singh et al. (2013)] that includes buildings, rivers, vegetation, bridges, etc. The building model is the key feature of the model.
The use of 3D city models is encouraged for a variety of reasons. The majority of 2D GIS use cases may theoretically be implemented with 3D GIS data. However, 3D city models may imitate realistic surroundings more precisely than 2D data, which increases the accuracy of the results and their interpretation [Biljecki et al. (2015)], such as in the urban wind flow simulation case [García-Sánchez et al. (2021)] and urban energy needs estimation [Agugiaro (2016); León-Sánchez et al. (2021)], these cases all confirm that 3D data could lead to a significant improvement of the estimations and assessments [Biljecki et al. (2015)]. The use of 3D city models for urban analysis is becoming more prevalent in contemporary research as traditional 2D GIS spatial data is no longer sufficient to suit the objectives of many studies. The current researcher has defined 12 categories of 3D city model use: emergency services, urban planning, telecommunications, architecture, facilities, and utility management, marketing and economic development, property analysis, tourism and entertainment, e-commerce, environment, education and learning, city portals [Batty et al. (2001)]. 3D city models can also be classified into non-visualization use cases (e.g., navigation) and visualization-based cases (e.g., virtual reality) [Biljecki et al. (2015)]. The abundance of non-visualization use cases suggests that the role of 3D models has gone beyond visualization to more development and utilization areas [Biljecki et al. (2015)], now their analytical capabilities are becoming more and more crucial.
One of the important characteristics of the 3D city model is the level of detail (LOD). LOD is a measure of how accurately a 3D city model has been created and how closely it adheres to the relevant subset of reality [Biljecki (2017)]. It is mostly used to characterize the geometric detail of a model, primarily of buildings, in the 3D GIS domain [Biljecki et al. (2016)]. The LOD between geographic data might vary depending on the nature of data, spatial scale, acquisition procedure, and other aspects [Biljecki et al. (2016)]. The detailed definition of LOD is shown in figure 1 and Table 3. It is worth noting that LOD3 is a lot more detailed model than LOD2 in this comparison. For many applications, the benefits presented in LOD 3 are quite helpful. The creation of a LOD3 3D city model is always a worthwhile subject to address because LOD3 can play a greater role in many spatial analyses, including illumination analysis and heat loss estimation.

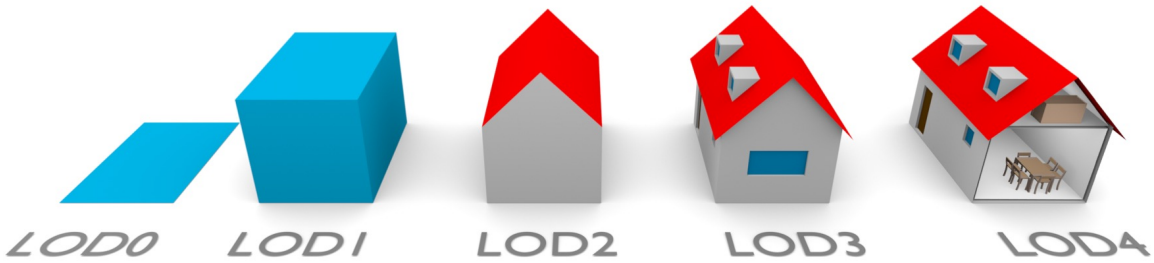

Figure 1: LOD specification [Biljecki et al. (2016)]

Table 1: LOD definition developed by OGC

LOD	Definition
LOD0	Buildings are represented by footprint or roof edge polygons.
LOD1	buildings are represented as blocks model, comprising prismatic buildings with flat roof structures.
LOD2	Buildings have differentiated roof structures and thematically differentiated boundary surfaces.
LOD3	Buildings have detailed wall and roof structures potentially including doors and windows.
LOD4	LOD4 buildings complete LOD3 buildings by adding interior structures.

The generation process of the LOD2 model has been developed to be relatively mature, and the existing photogrammetry-based methods and Laser Scanning based methods can basically implement the automatic reconstruction of the LOD2 model. The LOD3 building model, however, is more difficult to correctly and automatically reconstruct. The current LOD3 models are generated in the following methods, firstly by point clouds [Akmalia et al. (2014); Leberl et al. (2010)], which require very dense point clouds. Secondly by BIM models [Geiger et al. (2015)], and finally by combining multiple sources of data to extend the LOD2 model into LOD3 [Zhang et al. (2019); Gruen et al. (2019)]. In addition to point clouds, the images also contain a rich set of building-related information, such as building façades, and openings (including windows and doors). With the advancement of sensors and the platforms on which they are mounted, there are several airborne, terrestrial, and mobile image datasets available, such as street view images and airborne oblique images, from which it is possible to extract LOD3 model components like windows and doors. As a result, many researchers have explored using rich building information found in different imagery to construct LOD3 models, and this approach works well. But there are also drawbacks to the existing LOD 3 building model reconstruction technique, such as the need for manual labor throughout the process [Zhang et al. (2019); AlHalawani et al. (2013); Nan et al. (2010)], and the reconstruction is only effective for buildings with regular openings distribution [AlHalawani et al. (2013)], while the reconstruction method is not applicable when the distribution of the openings is irregular.
The LOD2 model is not always used to its full potential in the majority of the existing LOD3 reconstruction techniques, and LOD2 is not always extended to LOD3. So we consider: how can we use the data we already have the best? Can we create a more automated pipeline and use this information to create a new LOD3 building model, for instance, using tilt pictures with wide coverage and the resulting LOD2 building model? This research expects to propose a pipeline to apply parsing, segmentation, and extraction of openings on the façade using the

deep learning method based on oblique aerial images. Then optimize the result using a regularization algorithm and transform them into 3D openings and join the existing 3D city model, so that the openings are located in the correct position and become a hole on the façade. Since the 3D city models used in this research are well-established datasets, this research focuses more on automated segmentation, extracting, and the adaptation of 3d openings to existing 3d models to ensure that the converted 3d openings can be perfectly located on the 3D models.

2 Related work

This chapter aims to provide knowledge about extracting openings from images to implement LOD 3 building reconstruction. The chapter is organized as follows: chapter 3.1 gives knowledge about windows detection and extraction knowledge, 3.2 gives knowledge about Mask R-CNN, and chapter 3.2 gives knowledge about LOD3 building model reconstruction.

2.1 façade parsing, segmentation, and object extraction

The techniques for opening detection and extraction differ. The general concept is to provide additional data, including street view imagery and oblique aerial images, to improve the information on openings. Using openings edge information generated by semantic segmentation to enhance the outcomes produced by instance segmentation, Mao et al. suggested an integrated approach combining semantic segmentation with instance segmentation [Mao et al. (2022)]. The training dataset is enhanced in terms of data preparation by the inclusion of three annotations that take into account the spatial relationships between glass windows. The new aspect of this method is that numerous spatially connected glass panels are marked as one target as opposed to each sub-glass panel being annotated as a separate target. The best-performing Mask R-CNN was selected while training a machine learning model, such as segmentation. GSDNet and Transseg are two single semantic segmentation models that were chosen. The outcomes demonstrate the effectiveness of the integrated approach (Figure 3 (h)) have smoother edges and more complete interiors compared to the results of the single approach (Figure 3 (b, c, d, e, f)), with a segmentation accuracy of 98.85%. The shortcoming of this method is the poor performance in near-ground and small isolated glass façades because the façade is not rectified and there is more distortion in the near-ground area. The extracted windows cannot be added to the 3D city model directly.

Zhang et.al suggested a pipeline (figure 4) to automatically detect, interactively choose the type of façade elements, and add textures to the LOD2 model in order to produce LOD3 models that fit the urban simulation [Zhang et al. (2019)]. However, this method requires interaction to achieve reconstruction. There are methods that can assess the precision of openings extraction in addition to obtaining façade element information from imagery. In order to detect repeating patterns of façade elements and their deformation parameters, Halawani et al. suggested a semi-automatic framework [AlHalawani et al. (2013)], which can increase the accuracy of detection findings on façades with somewhat regular openings distribution. The limitation is the limited applicability, and the framework fails when the openings are irregularly distributed, or when there are significant light shading variations.

According to a pixel-level segmentation approach presented by Gadde et al., which is based on feature computation, stacked generalization learning is used to train a number of augmented decision tree classifiers. The segmentation results are continuously improved at each level, and the classifiers in later stages can rectify the classification faults in previous stages

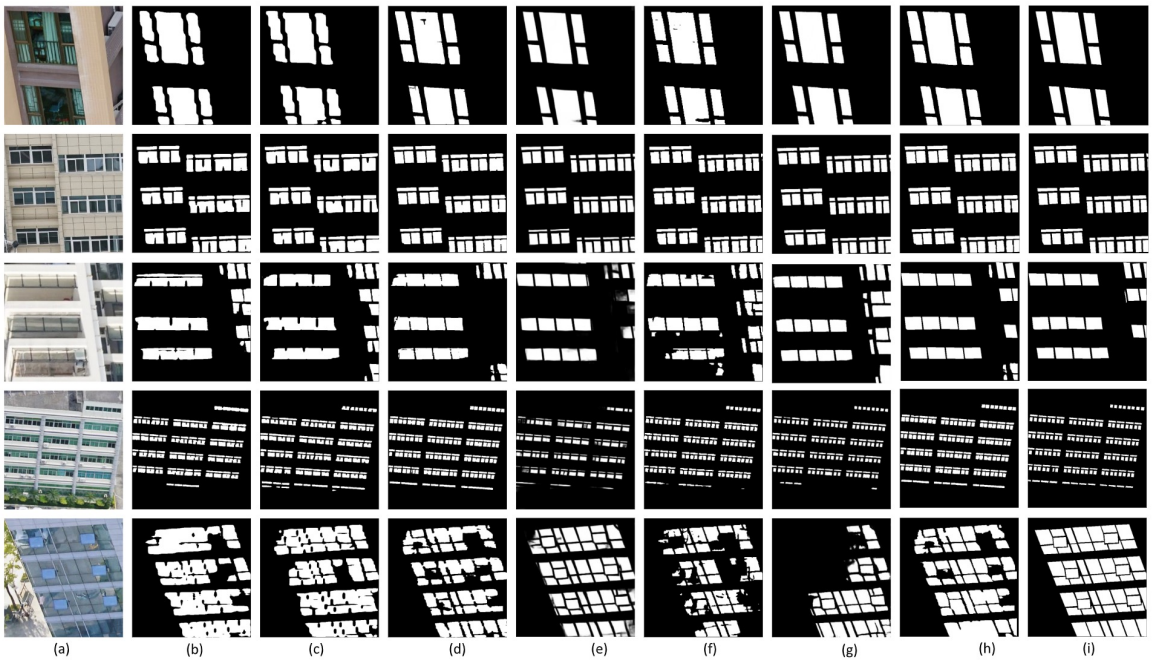

Figure 2: segmentation with different methods

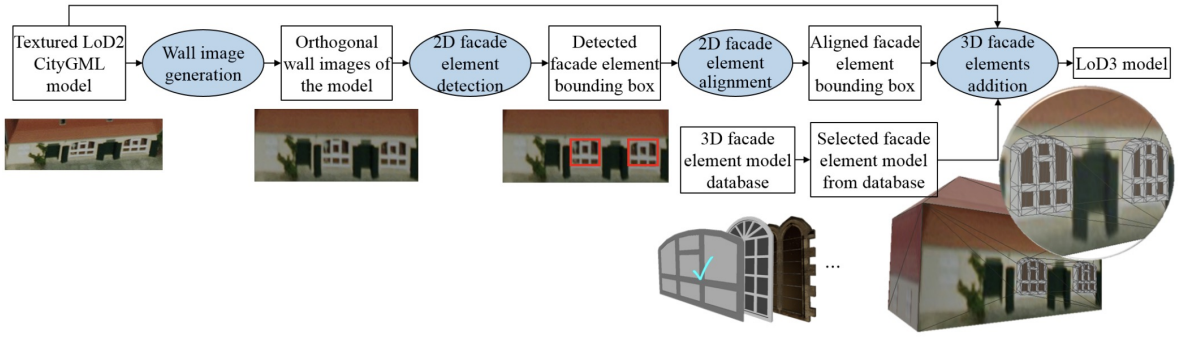

Figure 3: A Data-driven pipeline for Adding façade Details [Zhang et al. (2019)]

Figure 4: an extracted result [Zhang et al. (2019)]

[Gadde et al. (2018)]. This method is simple to use and adapt to new datasets, but because it mostly disregards existing knowledge, it classifies balconies and windows independently in the classification phase, leading to insufficient window segmentation.

2.2 Deep learning method

In LOD3 reconstruction techniques such as symmetry detection and façade segmentation, deep learning methodology is usually applied.
Based on the symmetry of the façade structure, Liu et.al proposed a symmetric regularizer called "Deepfaçade" for training the neural network [Liu et al. (2017)]. The basic idea is to train deep Convolutional neural networks with constraints under man-made rules: it uses a "symmetric loss term" based on the common symmetry found in structures like windows, walls, and doors to apply constraints or guide on the neural network.
Ren et.al proposed a Faster R-CNN, which introduced a Region Proposal Network (RPN) that shared full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals [Ren et al. (2015)]. An RPN is a fully-convolutional network that simultaneously predicts object bounds and objectness scores at each position.
Mask Region-based Convolutional Neuron Network (Mask R-CNN) is a popular deep learning model for instance segmentation, which is used for identifying and segmenting individual objects within an image [He et al. (2017)]. Mask R-CNN is an extension of Faster R-CNN, with the addition of a branch that predicts segmentation masks on each region of interest (RoI), in parallel with the existing branch for bounding box recognition. Mask R-CNN is known for its high accuracy in instance segmentation tasks, particularly when compared to other approaches that do not use deep learning. It is designed to be efficient at inference time, meaning that it can process images quickly and produce high-quality segmentation masks in real-time. Both Mask R-CNN and Faster R-CNN are popular deep-learning models for object detection and instance segmentation. While both models have their own advantages, they also have some key differences. Mask R-CNN produces object masks: In addition to identifying and bounding objects within an image, Mask R-CNN also produces a pixel-level mask for each object, which can be useful for tasks such as image segmentation or object manipulation, while Faster R-CNN only produces bounding boxes for objects. Mask R-CNN may be more accurate: In some cases, Mask R-CNN may produce more accurate object detections and segmentations than Faster R-CNN, particularly for tasks that require precise object masks. Due to the Mask R-CNN producing object masks in addition to bounding boxes, it is slightly slower than Faster R-CNN at inference time. However, this difference in speed is typically small and may not be noticeable in many applications.

2.3 LOD 3 building model reconstruction

Interactive detailed façades reconstruction has been well developed now. The SmartBoxes proposed by Nan et.al can form intelligent building blocks of 3D city models based on the input point clouds, and the user can manipulate the building blocks and complete the detailed reconstruction of façades and their elements Nan et al. (2010). Zhang et.al proposed to interactively select the matching façade element model from the 3D façade element database based on the detected façade elements [Zhang et al. (2019)]. The integration of 3D façade element models and 3d city models is achieved by transforming, positioning, and resizing. The above interactive 3D reconstruction approach improves the accuracy of the reconstruction and satisfies the user's need to obtain detailed surfaces, but to some extent, convenience and reconstruction efficiency is sacrificed. Wang uses the windows extracted from the images with known camera parameters to back-project the 2D windows onto the 3D façade [Wang (2022)], and modifies the size and location of the windows in the initial layout through final layout

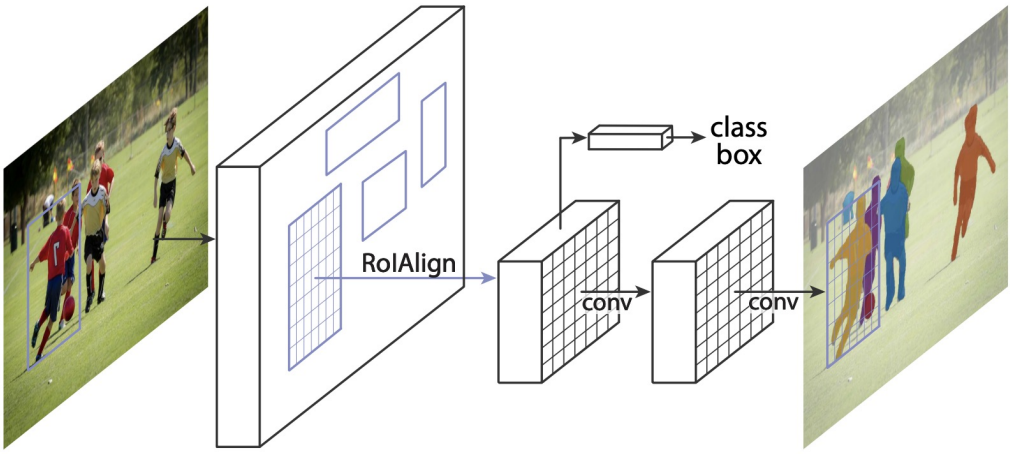

Figure 5: The general framework of Mask R-CNN He et al. (2017)

regularization, to improve the quality of the results. This method is very demanding on the input data because, for each reconstructed building, a large amount of image information is needed to implement the 3D reconstruction. And each image needs to provide camera internal parameters and external parameters, which makes this method relatively complicated in a larger range of LOD 3 model reconstruction. Additionally, this strategy only functions on buildings with a regular shape and façade; its effectiveness on complicated structures is unknown.

3 Research objectives

3.1 objectives

Based on the development status of the 3D City model and its limitations, the objective of this research is to propose a new pipeline to generate the LOD3 building model: by detecting and extracting openings from oblique aerial images and overlaying them on the LOD2 building model to generate the target result LOD3 model. The ideal pipeline should have as easy of a data preparation procedure as possible, and as automated of a reconstruction procedure as is practical. The ideal output LOD3 model should be that the openings are accurately positioned and that the openings fall exactly on the façade to form a set of holes. To achieve the objective, the relevant sub-questions are listed as follows:

- How can the manual component of the reconstruction process be minimized to optimize automated reconstruction?

- How can openings be extracted from images? How can the extracted results be evaluated?

- The challenge of reconstruction is somewhat increased for irregular façades. How to achieve ortho correction and automatic 3D transformation for irregular façades?

3.2 scope of research

The automatic extraction of openings from oblique aerial photos and the alignment of the 3D openings with the 3D city model will be the main topics of this study. This research therefore focuses on the pipeline's level of automation as well as the precision of size and position. The generation of the 3D city model is outside the purview of this research because it is based on the 3D BAG LOD2.2 model, which is an existing 3D city model.

3.3 research challenges

It can be difficult to extract openings from oblique aerial images. Oblique images do not have the same level of detail in 3D façade modeling as street view images. First of all, because oblique images are obtained at high altitudes and have very low resolution, there is a limited amount of information available for image segmentation. Second, most openings are made of glass, which has a certain amount of reflectivity and interferes with the findings of deep learning segmentation. façades in the real world are highly variable, and it is a great challenge to design programs that automatically adapt to diverse façades.

4 Methodology

4.1 Overview

This chapter provides a detailed explanation of the exact implementation strategy that will be applied for this research. Section 4.2 will illustrate the content related to façade detection, segmentation, and extraction, including data preparation, model use, and result accuracy verification. Section 4.3 will explain layout optimization, in terms of size and position, respectively. Section 4.4 will discuss the transformation between 2D façade and 3D façade. To sum up, the pipeline is divided into the following steps: (1) façade detection, segmentation, and extraction; (2) openings layout optimization; and (3) 3D openings transformation.

4.2 façade detection, segmentation, and extraction

Based on realistic research needs, the façade elements expected to be detected in this research include windows and doors. To maintain the realism of windows and doors, we will try to extract openings based on pixel-level segmentation results. This research expects to use Mask R-CNN for object instance segmentation. Mask R- CNN can effectively detect objects while generating a high-quality segmentation mask for each instance. The first stage of Mask R-CNN scans the image and generates proposals (areas likely to contain an object). It consists of two networks: backbone and region proposal network (RPN). These networks run once per image to give a set of region proposals. Region proposals are regions in the feature map that contain the object. In the second stage, the network predicts bounding boxes and object classes for each of the proposed regions obtained in the first stage. Each proposed region can be of a different size whereas fully connected layers in the networks always require fixed-size vectors to make predictions. The size of these proposed regions is fixed by using either the RoI pool (which is very similar to MaxPooling) or the RoIAlign method. The problem with Faster R-CNN is that the feature map is misaligned with the original image, which will affect the detection accuracy. The Mask R-CNN proposes the method of RoIAlign to replace ROI pooling, it computes the value of each sampling point by bilinear interpolation from the nearby grid points on the feature map, which can preserve the approximate spatial location.

4.2.1 Data collection

Firstly, the building corresponding to the façades to be extracted is determined according to the 3D BAG LOD0 data(footprint of the building), and the corresponding faces in the building (the index corresponding to the surface list in the CityJSON file). During the preliminary overview and experiments on the data, we found many kinds of complications. First, the occlusion may occur (e.g. due to the close proximity of the buildings). However, due to the presence of 30% side overlap and 60% forward overlap in the captured image, it is possible that the obscured building is extracted intact in the image next to it. We expect to ensure the completeness of façade extraction results in this way. Secondly, due to the complexity of the building structure, it is impossible to determine the number of façades to be extracted in advance, but as much information as possible will be extracted from each façade, as shown in Figure 6, we can extract information from multiple façades parts in one image. In addition to the uncertainty of the number of façades, there is also the complexity of shapes, and in buildings rich in design elements, façades often have irregular shapes. The approach taken in this study is to segment the irregular façades so that each sub-block becomes a regular shape 7.
Next, find the surface corresponding to each extracted façade in the 3D BAG LOD 2.2 building model. the true length and width of the surfaces will be used for façade image correction to obtain the orthoimages of the extracted façades.

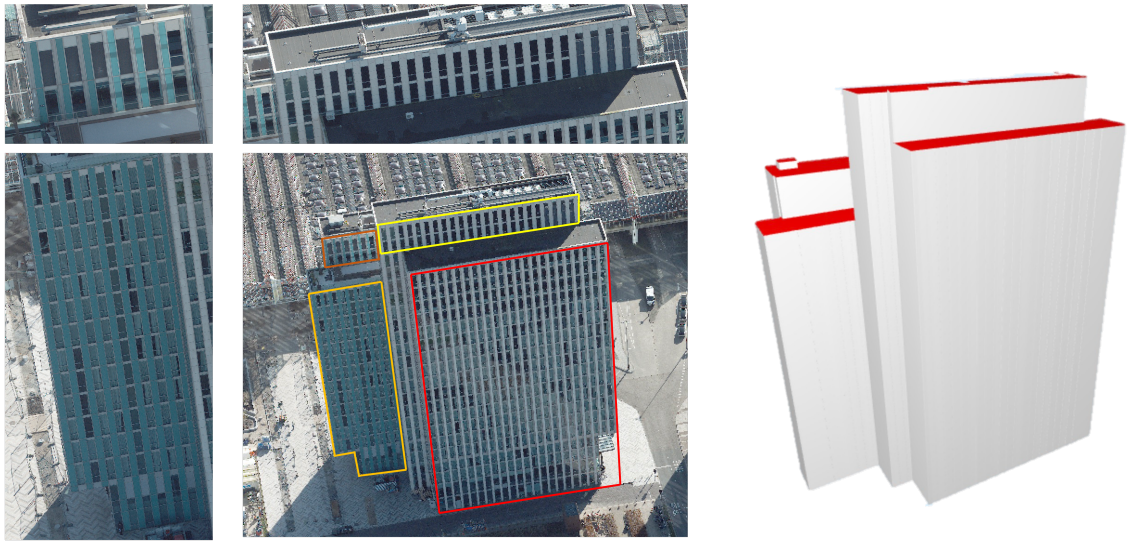

Figure 6: an example of complex building with multiple façades in one image (1314CH, Almere)

4.2.2 Data pre-pocessing

In this study, we intend to extract 100-200 façade images from oblique aerial imagery of the Almere area for Mask R-CNN training, of which 60% are used for training and 40% for testing. All the façade images will be resized to 512x512 pixels and applied standard normalization techniques to ensure that the pixel values are consistent across all images, to increase the diversity of the training data.

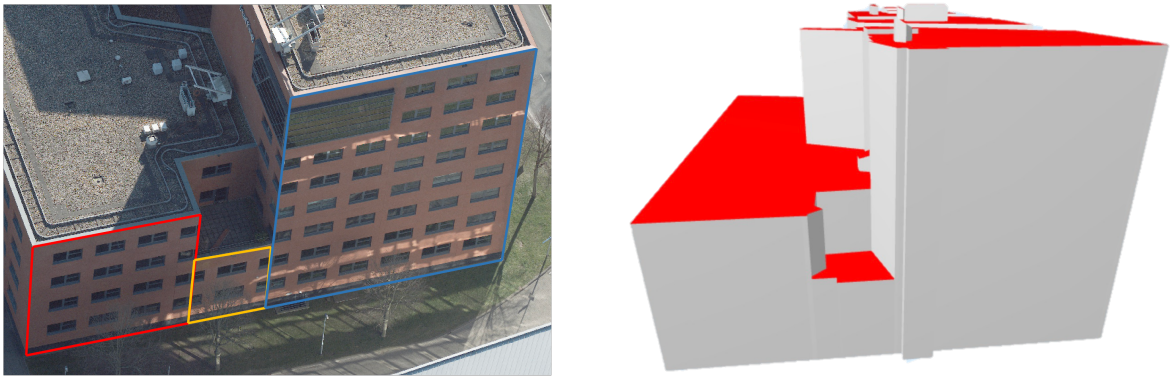

Figure 7: an example of building with irregular façade (1314CK, Almere)

4.2.3 Implementation of Mask R-CNN

For this research, we will use the Mask R-CNN model, which is a state-of-the-art method for instance segmentation. The model is based on the Faster R-CNN architecture and uses a region proposal network (RPN) to generate proposals for object instances, followed by a series of convolutional and fully connected layers to classify and segment each object.
To implement Mask R-CNN, we will use the open-source implementation provided by the Matterport team in the Mask R-CNN GitHub repository . This implementation is based on the PyTorch framework and includes pre-trained models on the COCO dataset, as well as tools for training and evaluating the model on custom datasets.
We will use a pre-trained Mask R-CNN model and fine-tune it on our dataset and use the Adam optimizer and a learning rate of 0.001, with a batch size of 2. We will train the model for 20 epochs and save the model with the lowest validation loss. To ensure the stability of our results, we will run the training process three times and report the average performance across all runs.
During training, we will use data augmentation techniques such as random horizontal flipping and random cropping to increase the diversity of the training data. We will also apply standard normalization techniques, such as mean subtraction and standard deviation scaling, to ensure that the pixel values are consistent across all images.
In the context of image segmentation, the loss function is used to measure the difference between the predicted segmentation masks and the ground truth masks. The goal of training a Mask R-CNN model is to minimize this difference so that the model can learn to produce accurate segmentation masks. Losses are evaluated using the loss function:

$$L = L_{cls} + L_{box} + L_{mask}$$

4.2.4 Evaluation

In this research, we provide some methods for assessing the performance of the Mask R-CNN model in terms of segmentation, including computing the IoU, precision, and recall.

- IoU: A typical statistic for assessing the effectiveness of object detection and segmentation algorithms. IoU can be used in the context of mask R-CNN to quantify the degree of overlap between the ground truth masks and the predicted masks generated by the model.

$$IoU = \frac{(mask_{predicted} \cap mask_{groundtruth})}{(mask_{predicted} \cup mask_{groundtruth})}$$

The value of IoU ranges from 0 to 1, with a higher value indicating a greater overlap between the predicted mask and the ground truth mask.

- Precision: The percentage of pixels that are accurately identified as being a component of the target object is measured by item precision. It is determined by dividing the total number of positive pixels predicted by the model by the number of actual positive pixels.
- Recall: The proportion of pixels in the ground truth mask that are properly identified as belonging to the target object is measured by recall. It is calculated by dividing the total number of pixels in the ground truth mask by the number of true positive pixels.

Obtaining the ground truth masks for the images being segmented is the initial step in calculating the precision and recall. These are the masks that represent the true locations of the objects in the images. Next, run the mask R-CNN model on the images and obtain the predicted masks. For each ground truth mask, compare it to the corresponding predicted mask. A predicted mask is considered a true positive if it overlaps significantly with the ground truth mask. The overlap can be measured using the IoU we calculated above. The precision is calculated as the number of true positive predictions divided by the total number of positive predictions made by the model, and the recall is calculated as the number of true positive predictions divided by the total number of actual positive samples in the dataset.

4.3 Layout optimization of the openings

When designing and constructing building façades, façade elements are typically arranged in a predictable pattern to increase aesthetics while preserving stability. For example, windows on the same floor of many buildings keep the same height, and windows on the same column remain vertical; in terms of size, the majority of windows are the same length and width. The outcome of the pixel-based segmentation approach may have some errors in its own shape and general layout due to the low resolution of oblique aerial photos. In order to remedy the problem, this research will enhance the extracted openings' quality by optimizing their size and position.
The constraints on openings in this study are set as follows:

- Windows that have comparable initial size should have the same width and length.
- Windows that are positioned normally (parallelly and vertically) should be changed to align horizontally and vertically.
- Doors and windows should have a horizontal width and a vertical height.

4.3.1 Size regularization

The purpose of size regularization is to adjust the size of windows with similar lengths and widths, to avoid errors caused by the segmentation process, and to reduce the size differences between windows that are supposed to be the same. Therefore, in this stage, we will implement automatic window resizing through two rounds of clustering.
1. To minimize manual participation in the pipeline, this study takes advantage of the fact that hierarchical clustering does not require a predetermined number of clusters to predict the types of windows.
2. Combining the number of types obtained from hierarchical clustering, windows that originally have the same widths and lengths are classified again using k-means clustering. Calculate the average of the length and width of similar windows $\bar{x_m}$ and $\bar{y_m}$ while m is the

clustering result, and replace the original x_i and y_i with the average. Keep the coordinates of the upper left corner of the window unchanged and resize the window by changing the coordinates of the other three corners (see Figure 8).

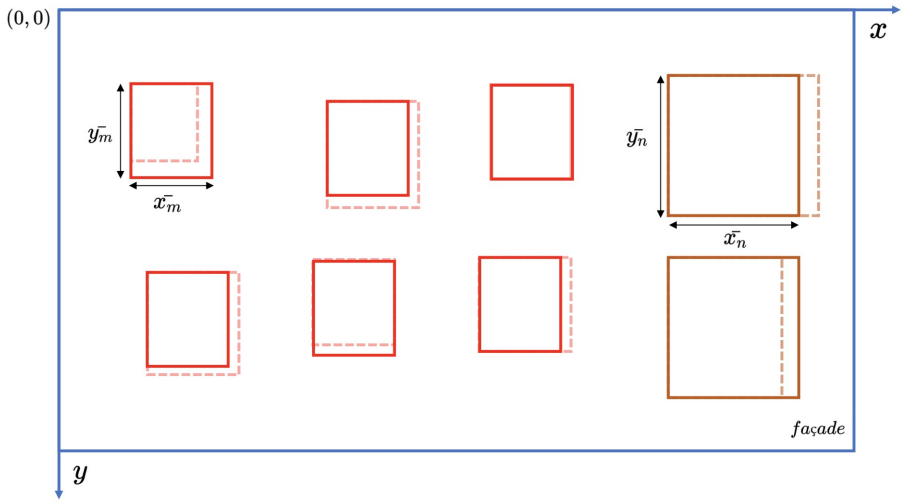

Figure 8: constraint of size

4.3.2 Position regularization

Theoretically, most windows are aligned horizontally and vertically with their neighboring windows. To maintain this limitation, some of the windows need to be moved slightly to maintain the correct relative position. For each window, the centroid can be calculated to represent this window.

For each row of relatively horizontal windows, we will adjust the y-value of its coordinates to transform all windows to the same average height. The average y-value $y = \bar{y_n}$ of all the windows is calculated, then the transforming distance in the y-direction is y_i - $\bar{y_n}$, and the y-value of the transformed centroid is $\bar{y_m}$. Correspondingly, for each column of relatively vertical windows, adjust the x-value of their coordinates, transforming to the same vertical line $x = \bar{x_m}$, with the value calculated by averaging the x-values of all centroids. The transforming distance in the y-direction is y_i - $\bar{y_n}$, and x-value of the transformed centroid is $\bar{x_m}$.

To summarize, the coordinates of the transformed center point of the window located in column m and row n are ($\bar{x_m}$, $\bar{y_n}$), all windows will be moved with the center position to achieve the result of window alignment (see Figure 9).

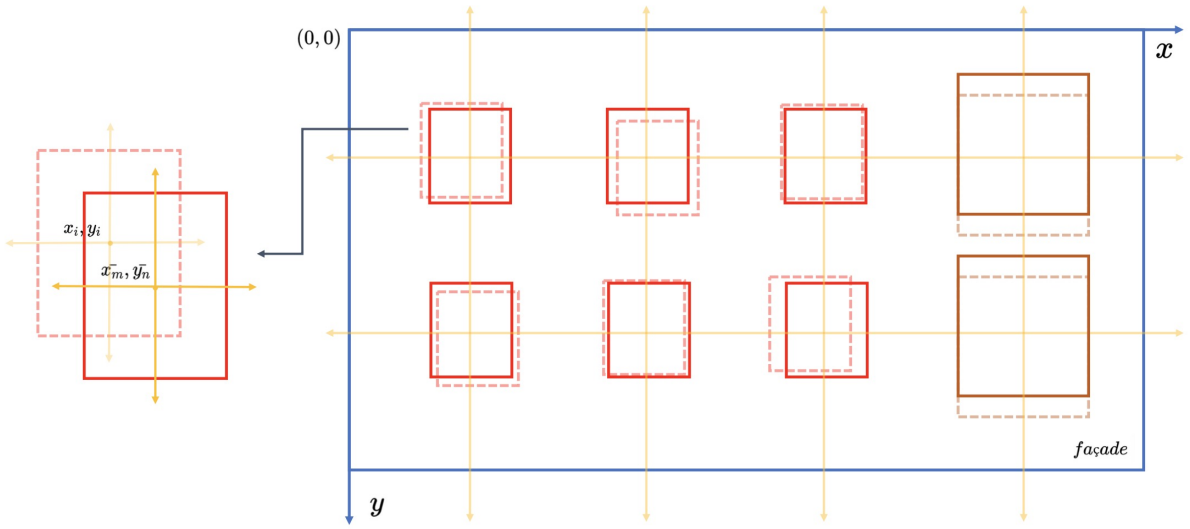

Figure 9: constraint of position

4.4 3D openings transformation

The last stage is to transform the 2D façade elements into 3D ones after acquiring the correct distribution of façade elements. Instead of calculating the 3D coordinates using the 3D reconstruction approach because we already have ready-made 3D building models, this research combines the coordinates of the 3D façades to derive the 3D coordinates of the façade elements in order to assure the optimal integration of the two. The basic idea of this approach is that each façade may be viewed as a 3D plane with associated plane functions. The 3D coordinates of each corner can be calculated by replacing certain values if we can locate the plane in which the corner of the façade element is positioned.
We consider each surface to be continuous and whole (assuming this problem will be fixed by 3DGI). Calculating the 3D coordinates for the corners of apertures for each façade (for example, façades are rectangles) and their related 3D surfaces will be quite simple. The relative positions of corners in 2D can be used to determine the 3D coordinates in the 3D plane so that the generated 3D points fall on the plane. Figure 10 illustrates how this technique might be put into practice.

$$x = \frac{x_{2D}}{X_{2D}} \times X_{3D}$$
$$y = \frac{x_{2D}}{X_{2D}} \times Y_{3D}$$
$$z = \frac{z_{2D}}{Z_{2D}} \times Z_{3D}$$

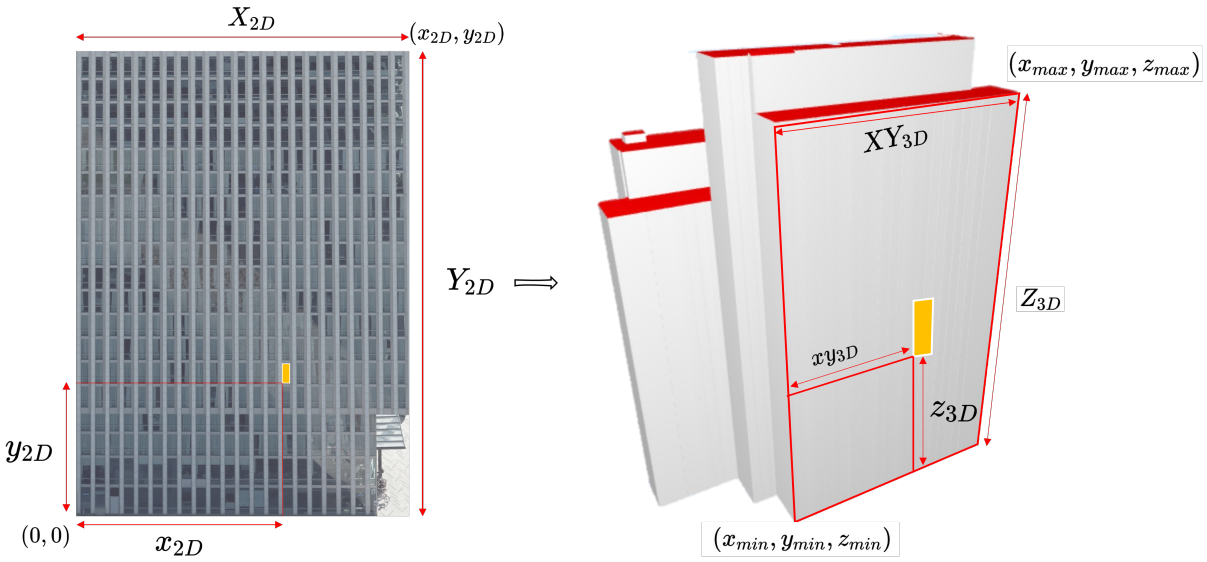

Figure 10: 3D coordinates transformation

5 Time planning

The expected timeline of this study is as follows:

Table 2: Time schedule

date	task
15 Oct - 31 Oct	literature review
1 Nov - 15 Nov	P1 MSc thesis topic submission
30 Nov - 15 Dec	Literature review
15 Dec - 24 Dec	Write the literature review of the research proposal
1 Jan - 15 Jan	Write the methodology part of the research proposal
23 Jan	P2 presentation
1 Feb - 6 Feb	Research data preparation
6 Feb - 10 Feb	Spring Break
13 Feb - 15 Feb	Research data preparation
16 Feb - 5 Mar	Work on Mask R-CNN training & implementation
6 Mar - 15 Mar	Model evaluation & model optimizing
16 Mar - 7 Apr	Work on façade layout optimization
10 Apr - 30 Apr	Work on 3D openings transformation
1 May - 15 May	Write thesis
17 May	P4 presentation
20 May - 20 Jun	Finalize the MSc thesis and prepare for the final presentation

6 Tools and datasets used

6.1 research area and dataset

The experiments in this research project were performed in the Almere area, and the oblique aerial images for the experimental data were provided by Gemeente Almere, and the data collectors were CityMapper-2 airborne sensor systems. The dataset includes images taken from

four directions, forward-looking, back-looking, left-looking, and right-looking. The aircraft altitude was 1475 ft, resulting in he resolution of the oblique photographs was between 1.9 (short side) and 2.7 cm (far side).
The area of focus of this research proposal is the buildings near Almere Centrum.

Table 3: sensor information

camera parameter	value
side oblique pixels across	10640
side oblique pixels along	13192
fwd/back oblique pixels across	14192
fwd/back oblique pixels along	10640
oblique focal length	112mm
oblique angle	45°
Field of view	40° conical scan
system coordinates	RDNAP

The 3D building model is 3D BAG LOD 2.2 data, it is an up-to-date data set containing 3D building models of the Netherlands. The data can be obtained from https://3dbag.nl/en/download.

6.2 opencv-python

Opencv-python package is used for façade image correction in section 4.2.1.

6.3 COCO Annotator

COCO Annotator is a web-based image annotation tool for creating training datasets for Mask R-CNN. The export annotations in COCO format can be used in Mask R-CNN directly. In this research, COCO Annotator will be used for segmentation and annotation of façades and openings on oblique images, and exported for training Mask RCNN.

6.4 Mask R-CNN

Mask R-CNN is the main tool for extracting openings from oblique images in this experiment. It relies on an environment that includes: numpy, scipy, Pillow, cython, matplotlib, scikit-image, tensorflow (1.3.0), keras (2.0.8), opencv-python, h5py, imgaug.

6.5 Ninja

The visualization tool of 3D BAG data is Ninja.

References

G. Agugiaro. Energy planning tools and citygml-based 3d virtual city models: experiences from trento (italy). *Applied Geomatics*, 8(1):41–56, 2016.

R. Akmalia, H. Setan, Z. Majid, D. Suwardhi, and A. Chong. Tls for generating multi-lod of 3d building model. 18(1):012064, 2014.

S. AlHalawani, Y.-L. Yang, H. Liu, and N. J. Mitra. Interactive facades analysis and synthesis of semi-regular facades. 32(2pt2):215–224, 2013.

M. Batty, D. Chapman, S. Evans, M. Haklay, S. Kueppers, N. Shiode, A. Smith, and P. M. Torrens. Visualizing the city: communicating urban design to planners and decision-makers. 2001.

F. Biljecki. Level of detail in 3d city models. 2017.

F. Biljecki, J. Stoter, H. Ledoux, S. Zlatanova, and A. Çöltekin. Applications of 3d city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889, 2015. ISSN 2220-9964. doi: 10.3390/ijgi4042842. URL `https://www.mdpi.com/2220-9964/4/4/2842`.

F. Biljecki, H. Ledoux, and J. Stoter. An improved lod specification for 3d building models. *Computers, Environment and Urban Systems*, 59:25–37, 9 2016. ISSN 01989715. doi: 10.1016/j.compenvurbsys.2016.04.005.

R. Gadde, V. Jampani, R. Marlet, and P. V. Gehler. Efficient 2d and 3d facade segmentation using auto-context. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(5):1273–1280, 2018. doi: 10.1109/TPAMI.2017.2696526.

C. García-Sánchez, S. Vitalis, I. Paden, and J. Stoter. The impact of level of detail in 3d city models for cfd-based wind flow simulations. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 46(4/W4-2021), 2021.

A. Geiger, J. Benner, and K. H. Haefele. Generalization of 3d ifc building models. pages 19–35, 2015.

A. Gruen, S. Schubiger, R. Qin, G. Schrotter, B. Xiong, J. Li, X. Ling, C. Xiao, S. Yao, and F. Nuesch. Semantically enriched high resolution lod 3 building model generation. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:11–18, 2019.

K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

F. Leberl, A. Irschara, T. Pock, P. Meixner, M. Gruber, S. Scholz, and A. Wiechert. Point clouds: Lidar versus 3d vision. *Photogrammetric Engineering Remote Sensing*, 76(10):1123–1134, 2010. doi: 10.14358/pers.76.10.1123.

C. León-Sánchez, D. Giannelli, G. Agugiaro, and J. Stoter. Testing the new 3d bag dataset for energy demand estimation of residential buildings. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 46(4/W1-2021), 2021.

H. Liu, J. Zhang, J. Zhu, and S. C. Hoi. Deepfacade: A deep learning approach to facade parsing. IJCAI, 2017.

Z. Mao, X. Huang, Y. Gong, H. Xiang, and F. Zhang. A dataset and ensemble model for glass façade segmentation in oblique aerial images. *IEEE Geoscience and Remote Sensing Letters*, 19: 1–5, 2022.

L. Nan, A. Sharf, H. Zhang, D. Cohen-Or, and B. Chen. Smartboxes for interactive urban reconstruction. 2010. doi: 10.1145/1833349.1778830. URL `https://doi-org.tudelft.idm.oclc.org/10.1145/1833349.1778830`.

R. Peters, B. Dukai, S. Vitalis, J. van Liempt, and J. Stoter. Automated 3d reconstruction of lod2 and lod1 models for all 10 million buildings of the netherlands, 2022. ISSN 0099-1112.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

S. P. Singh, K. Jain, and V. R. Mandla. Virtual 3d city modeling: Techniques and applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-2/W2:73–91, 2013. doi: 10.5194/isprsarchives-XL-2-W2-73-2013. URL `https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-2-W2/73/2013/`.

L. Wang. Detailed facade reconstruction for mahattan-world buildings. 2022. URL `http://resolver.tudelft.nl/uuid:1581231d-6109-44b2-ab57-bbbea3e08642`.

X. Zhang, F. Lippoldt, K. Chen, H. Johan, M. Erdt, X. Zhang, F. Lippoldt, K. Chen, H. Johan, and M. Erdt. A data-driven approach for adding facade details to textured lod2 citygml models. pages 294–301, 2019.