

SEMI-AUTOMATIC GENERATION OF AN LOD1 AND LOD2 3D CITY MODEL OF TANAUAN CITY, BATANGAS USING OPENSTREETMAP AND TAAL OPEN LIDAR DATA IN QGIS

P.H.T. Camacho^{*1}, V.M.R. Santiago¹, C.J.S. Sarmiento¹

¹ Department of Geodetic Engineering, University of the Philippines,
Diliman, Quezon City, 1101, Philippines – (ptcamacho, vrsantiago, cssarmiento)@up.edu.ph

KEY WORDS: 3D City Model, Building Reconstruction, LOD1, LOD2, LiDAR, OpenStreetMap, FOSS, QGIS.

ABSTRACT:

3D city models have found purpose beyond simple visualization of space by serving as building blocks of digital twins and smart cities. These are useful to urban areas in the Philippines through diversified applications: urban planning, disaster mitigation, environmental monitoring, and policy making. This study explored the use of free and open-source software to generate an LOD1 and LOD2 3D city model of Tanauan City, Batangas using building footprints from OpenStreetMap and elevation models from Taal Open LiDAR data. The proposed approach consists of GIS-based methods including data pre-processing, building height extraction, roof identification, building reconstruction, and visualization. The study adopted methods from previous studies for building detection and from Zheng et al. (2017) for LOD2 building reconstruction. For LOD1, a decision tree classifier was devised to determine the appropriate height for building extrusion. For LOD2, a model-driven approach using multipatch surfaces was utilized for building reconstruction. The workflow was able to reconstruct 70.66% LOD1 building models and 55.87% LOD2 building models with 44.37% overall accuracy. The RMSE and MAE between the extracted heights from the workflow and from manual digitization has an accuracy lower than 1 m which was within the standards of CityGML. The processing time in test bench 1 and test bench 2 for LOD1 took 00:12:54.5 and 00:09:27.2 while LOD2 took 02:50:29.1 and 01:27:48.2, respectively. The results suggest that the efficient generation of LOD1 and LOD2 3D city models from open data can be achieved in the FOSS environment using less computationally intensive GIS-based algorithms.

1. INTRODUCTION

A 3D City Model is a simplified virtual representation of an urban environment for the mathematical calculations and predictions of the systems and processes in that area. The accelerated evolution of modern computing power has allowed three-dimensional digital spatial analysis and modelling to be more common and accessible as compared to the traditional two-dimensional counterpart. Three-dimensional models give vertical information a more intuitive visualization and perspective.

CityGML is the international standard adopted by the Open Geospatial Consortium (OGC) for the representation, storage, and exchange of 3D City Models (Gröger et al., 2012). It differentiates multiple representations of a 3D city model through five consecutive Levels of Detail (LOD) based on its geometric and semantic complexity (Arroyo Otori et al., 2018). Biljecki et al. (2014) define LOD as the model's degree of adherence to its corresponding counterpart such that increasing the LOD, increases the quality, accuracy, and complexity of the model. The LOD also reflects the quality of data acquisition and specific application requirements for efficient visualization and data analysis (Gröger et al., 2008).

A building model may be represented in different LODs from LOD0 to LOD4. LOD0, the coarsest level, is a two and a half dimensional polygon (Gröger et al., 2012) with its roof level height as an attribute (Biljecki et al., 2013). Models in LOD0 mark the transition from two dimensional (2D) to three dimensional (3D) GIS and thus contain no volume. LOD1 is a generalized block model with a flat roof (Gröger et al., 2008) obtained by extruding the building footprint or LOD0 model to

a uniform height (Arroyo Otori et al., 2018). LOD2 is essentially an LOD1 model enhanced with a generalised roof form as well as roof superstructures (Arroyo Otori et al., 2018) such as chimneys and dormers (Biljecki et al., 2013). LOD3 is the most complex and most detailed representation of the outermost form of a building (Gröger et al., 2008). Lastly, LOD4 completes the LOD3 model with the addition of indoor structures such as rooms, interior doors, stairs, and furniture (Gröger et al., 2008).

Several studies focused on automating or semi-automating the 3D reconstruction of buildings highlighted the importance of identifying and establishing the basic geometric unit for the reconstruction mechanism, which also defines the appropriate modelling process (Fu et al., 2008). With this, the problem of building reconstruction is simplified into the subsequent process of building region identification and geometric reconstruction based on established strategies or approaches (Rottensteiner et al., 2014 in Awrangjeb et al., 2013; Zheng et al., 2017; Chen et al., 2006).

In general, these approaches can be categorized into two major strategies: data-driven and model-driven, however, an occasional hybrid of the two was also demonstrated in some studies (Zheng et al., 2017). The data-driven approach, also known as the non-parametric (Sajadian et al., 2014), generic, or polyhedral approach (Lafarge et al., 2010 and Satari et al., 2012 in Awrangjeb et al., 2013) is identified as a bottom-up strategy (Chen et al., 2006) owing to its method of modelling which prioritizes basic principal structures or components that make up the whole building. For this approach, the reconstruction is focused on the detection and decomposition of the roof structures in point cloud data, which means that the points are segmented

into planes characterizing these structures (Tasha-Kurdi et al., 2019; Awrangjeb et al., 2013). Meanwhile, the model-driven approach, also known as the parametric approach (Awrangjeb et al., 2013), is defined as a top-down reconstruction strategy, which means that the modelling process is anchored on a hypothetical building framework (Chen et al., 2006). Being parametric in nature, this approach makes use of a model library containing pre-defined synthetic building primitive models which are tested for fit on the input data based on different roof type detection and fitting algorithms (Zheng et al., 2017; Dorninger et al., 2008).

This study aims to create a workflow which can efficiently and semi-automatically generate an LOD1 and LOD2 3D city model of Tanauan City, Batangas using open data available in the Philippines and the free and open source software (FOSS), QGIS, for processing. The use of open data and free and open source software (FOSS) will allow for replicability, reproducibility, and interoperability of 3D city models to maximize their potential in representing the complexities of urban areas. This will open the doors for more opportunities for engagement, participation, and collaboration of citizens in developing solutions for smart city initiatives. The success of the developed workflow for this research will greatly contribute to the development of 3D geoinformation in the country, most especially for different smart cities applications.

2. STUDY AREA, DATA SOURCES, & SOFTWARE

2.1 Study Area

The choice of study area is limited within the region near the Taal volcano in consideration of the availability of the Taal Open LiDAR Data. The selected study area is a 4000m x 4000m urban area located in Tanauan City, Batangas. It is encompassed by four adjacent tiles from the Taal Open LiDAR dataset: tiles E299N1558, E299N1559, E300N1558, and E300N1559.

2.2 Data Sources

This study explored the use of open data in the Philippines. Two main data sources were utilized: OpenStreetMap (OSM) data files from Geofabrik and Taal Open LiDAR data from the DREAM and Phil-LiDAR Programs. The datasets obtained are freely available online and accessible to the public.

2.3 Software

This study was implemented using free and open-source software namely QGIS and its pre-installed packages such as GDAL and SAGA GIS, Python for data processing, and the Qgis2threejs plugin for visualization.

3. METHODOLOGY

For this study, the advantages of the workflow presented in Zheng et al. (2017) provided the motivations for developing a method for semi-automating the building reconstruction using building footprints and LiDAR nDSM.

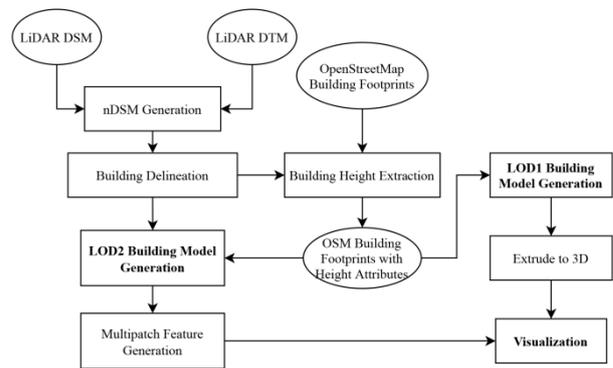


Figure 1. General workflow for generating the LOD1 and LOD2 3D city models.

The general workflow for the LOD1 and LOD2 3D city models is shown in Figure 1. The process is divided into 4 main parts which consist of Building Delineation, Building Height Extraction or LOD1 Building Model Generation, LOD2 Building Model Generation, and Visualization.

3.1 LOD1 Building Model Generation

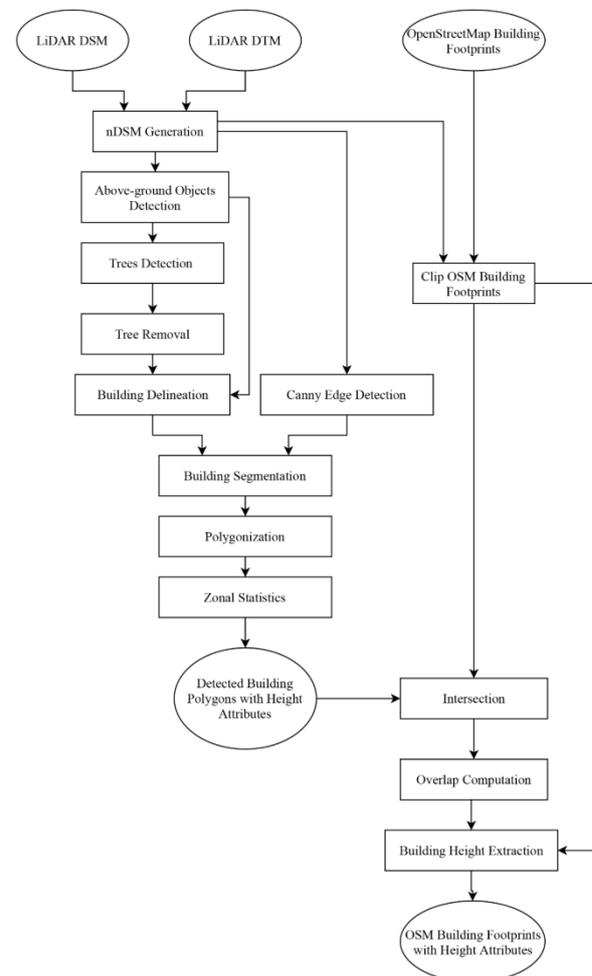


Figure 2. LOD1 Building Reconstruction Flowchart.

For the LOD1 Building Model Generation, the more detailed workflow is shown in Figure 2. To extract the appropriate heights for the OSM building footprints, building features were delineated and extracted from the LiDAR data through the

generation of the nDSM with 1 meter resolution. The presence of trees in urban areas similar to the study area poses difficulties in delineating building features since it complicates the height information extraction especially in areas where trees may occlude buildings. To avoid this, the study adopted the methods of Martin et al., 2014, Mendes et al., 2011, and Villanueva et al., 2015 to remove trees and other non-building features from the elevation model. Tree features were detected using the Terrain Ruggedness Index which was filtered using raster reclassification. Morphological filters were applied to the reclassified texture raster. The resulting raster containing the building features was segmented using the detected edges from the Canny Edge Detection algorithm applied to the nDSM. The segmented buildings raster was then polygonized. The mean and maximum heights for the segmented polygons were extracted using the Zonal Statistics tool. The resulting polygons with height attributes were intersected with the OSM building footprints and the overlap percentage was computed.

A decision tree classifier was devised to carefully choose the appropriate building height to be extracted depending on the overlap cases. There were six cases considered, four of which extracted heights for the building footprints, as summarized in Table 1. The percent overlap threshold value used in the classification was equal to 20%. This was based on the judgement of the authors as observed from the positional offset of the OSM building footprints and was assumed as the safest value that can extract appropriate heights to most, if not all the building footprints. The building footprints were then extruded using the Qgis2threejs plugin using the extracted heights to generate the LOD1 3D city model.

	Height	Assumption
1:1 (OSM building footprint overlaps with only one nDSM building)		
Case 1	Mean height of overlapping nDSM building	Area overlap of the nDSM building met the threshold and is therefore a good representative of the building.
Case 2	No building height	Area overlap did not meet the threshold and therefore is not a good representative of the building. Difference in date of acquisition shall be considered as some buildings from LiDAR may have been replaced with new ones.
1: many (OSM building footprint overlaps with more than one nDSM building)		
Case 3	Mean height of nDSM building with maximum overlap	If all overlapping nDSM buildings meet the threshold, then they are assumed to be probable step edges.
Case 4	Mean height of nDSM building with maximum overlap	If not all overlapping nDSM buildings meet the threshold, then the one with the maximum overlap shall be considered.
Case 5	Mean height of nDSM building with highest value	If no overlapping nDSM building has met the threshold and the height difference is less than 1 storey, then the one with the highest mean height shall be considered.

Table 1. The decision tree classifier for building height extraction.

3.2 LOD2 Building Model Generation



Figure 3. LOD2 Building Reconstruction Flowchart.

The separate workflow for extending the building models from LOD1 to LOD2 is shown in Figure 3. The workflow consists of Step-edge Detection, Roof Type Identification, and Building Reconstruction.

Similar to the building features segmentation in the LOD1 workflow, Canny Edge Detection was used to extract edges from the nDSM. Building footprints were segmented using the same method in the previous workflow. Subsequent step-edge refinement was done using the method given in Zheng et al. (2017) which utilized the Minimum bounding geometry and bend simplification method. The output of the step-edge segmentation and refinement was to be supposedly used in the next steps.

To identify the roof types of the building footprints, the aspect and slope maps of the nDSM were computed. The aspect values were reclassified based on the orientation values given by Zheng et al. (2017). The slope map was reclassified based on the assumption that flat roofs have slope less than or equal to 10 degrees. Both reclassified maps were polygonised and used in roof type extraction.

In this study, four roof types were evaluated: flat, gable, hipped, and complex roof. The attributes stored in the polygonized aspect and slope maps were used in deriving the percent area of the plane and its orientation value. Vector-based methods were utilized such as intersection and join attributes by location to process the candidate roof planes based on the osm id of the intersecting building footprint. The characteristics of each roof type was used in the extraction algorithm such as the number of planes, the percent by area, and the orientation.

The roof type extracted for each osm building footprint was stored as an attribute and used in the model-driven building reconstruction. For this, the multipatch generation function in PyShp was used. The 3D coordinates of the vertices of an OSM building footprint were extracted in QGIS and were processed in PyQGIS. The reconstruction utilized triangle strip and triangle fan multipatch surfaces. The principal axes of the buildings were estimated by computing the bearing of each polygon segment.

Four (4) building footprints shapes were reconstructed for gable roofs while the two shapes were reconstructed for hipped roofs. Eave heights were assumed to be the mean height extracted. The ridge height for the 3D roofs was assumed to be the maximum height while the length of the ridge line was only estimated. The generated multipatch features were visualized using the Qgis2threejs plugin and the flat-roofed buildings were extruded.

In Villanueva et al. (2015), the semi-automation of the workflow for extracting building footprints from LiDAR data using open-source GIS softwares like QGIS through python scripting and QGIS plug-in development was recommended after successfully implementing the methods through the separate tools available in the FOSS used. For this study, the semi-automation of the tools and methods for LOD1 building height extraction and roof type extraction were done through the Graphical Modeler tool of QGIS. Meanwhile, the height extraction, roof type extraction, and multipatch creation algorithms were executed through python scripting. The individual graphical models and python scripts were designed to automatically take in outputs from previous models or scripts as inputs based on the specific file names assigned to them, except for the first graphical model which requires the user to select the input DEMs and the OSM building footprints.

4. RESULTS AND DISCUSSION

4.1 LOD1 Building Model Generation

Using the height extraction workflow developed in this study on the 6,633 OSM building footprints found in the four (4) LiDAR tiles within the study area boundary, 4,687 or 70.66% were assigned with heights.

To assess the vertical accuracy of the heights extracted from the LiDAR nDSM, test buildings identified from the subset study area were manually digitized using georeferenced Google Earth images. Tile E300N1558 was chosen as the subset study area since it has the greatest number of building structures within its extent, as shown in Figure 4.

For the accuracy assessment, 504 test buildings were identified and the heights extracted were used as the reference heights. The heights of the test buildings were extracted from the same LiDAR nDSM used through the zonal statistics tool. Since the heights extracted will be used as the reference heights, the manually digitized footprints were assumed to be the accurate representation of the buildings on the ground. Thus, the assessment was based on the difference in heights extracted for the OSM building footprints and the manually digitized footprint caused by the positional offset and the discrepancies in the building footprints shapes.



Figure 4. Manually digitized test buildings and the corresponding OSM building footprints.

The descriptive statistics computed and used for height accuracy assessment are the root-mean-square error (RMSE), mean error (ME), mean absolute error (MAE), and standard deviation (SD).

In this study, the height extraction workflow was able to estimate the mean heights of the buildings at an accuracy lower than 1 m as shown by the values of the MAE and the RMSE. Considering

the LiDAR accuracy of ± 0.2 m (Paringit et al., 2017), the mean heights extracted were within the absolute vertical accuracy of 5m and 2m set by OGC for CityGML for both LOD1 and LOD2, respectively.

The per case height accuracy assessment showed that Case 3 poorly extracted accurate mean heights with respect to the LiDAR accuracy as compared to the other cases while Case 1 produced the least RMSE and MAE, as shown in Table 3.

Height Extracted	RMSE	ME	MAE	SD	R ²
Mean	0.650	0.074	0.410	0.646	0.937
Maximum	1.759	-0.510	0.921	1.685	0.733

Table 2. Descriptive Statistics for Mean and Maximum Height Extraction.

Case No.	No. of Test Bldgs	RMSE	ME	MAE	SD	R ²
1	229	0.424	0.090	0.298	0.415	0.948
3	26	1.364	-0.597	0.992	1.250	0.702
4	219	0.704	0.070	0.437	0.702	0.952
5	30	0.699	0.570	0.570	0.412	0.737

Table 3. Descriptive Statistics for Mean Height Extraction (Per Case).

Case No.	No. of Test Bldgs	RMS E	ME	MAE	SD	R ²
1	229	1.407	-0.191	0.665	1.397	0.656
3	26	2.454	-1.398	1.844	2.056	0.464
4	219	2.068	-0.844	1.151	1.892	0.759
5	30	0.652	0.263	0.396	0.607	0.605

Table 4. Descriptive Statistics for Maximum Height Extraction (Per Case).

The maximum height validation results show that the accuracy requirement for LOD2 can be achieved using the extracted maximum height. Based on the per case analysis, heights extracted under Case 3 were the least accurate among others, both for mean and maximum heights. Meanwhile, Case 1, which performed better in both the mean and maximum height extraction, demonstrates the effectiveness of the 20% area threshold in the development of the workflow.

Errors in the extracted heights can be attributed to the over-segmentation and under-segmentation of the nDSM detected buildings, which was dependent on the quality of the Canny Edge Detection output. The algorithm failed to detect some edges which caused the workflow to fail in properly segmenting closely spaced buildings. However, it was also observed that the under-segmented blocks in the nDSM detected buildings were mostly of the same building heights/levels and are mostly found in residential blocks (Figure 5). On the other hand, since the maximum height extraction gets the elevation of the highest point, it was observed to be more sensitive to under-segmentation than the mean height extraction since adjacent buildings have different specific highest points.



Figure 5. Areas showing under-segmentation of detected nDSM buildings.

Over-segmentation of nDSM buildings was also observed (Figure 6). For some cases of over-segmentation, the workflow opted to not extract heights to avoid the risk of assigning erroneous heights to the OSM building footprints.

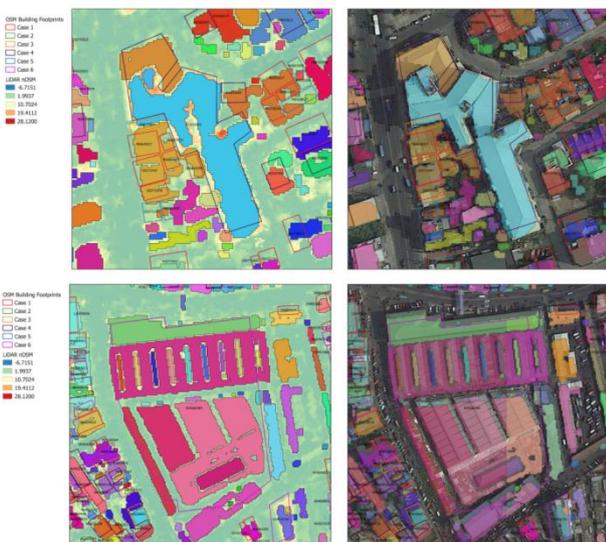


Figure 6. Areas showing over-segmentation of detected nDSM buildings.

The extraction method employed in Case 1, as well as Case 4, was determined to have performed better in areas with low building density and well-distributed open spaces (Figure 7). This can be attributed to the performance of the building detection and segmentation workflow.

The visualized LOD1 3D building models are shown in Figures 8 and 9 with the LiDAR DTM as the base map draped with Bing Maps using the Qgis2threejs plugin.

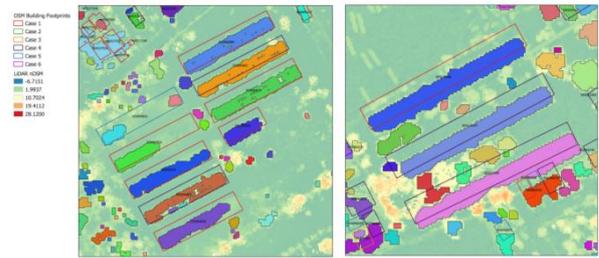


Figure 7. Areas showing buildings under Case 1 and Case 4.



Figure 8. Panoramic view of the LOD1 3D City Model generated using mean heights.

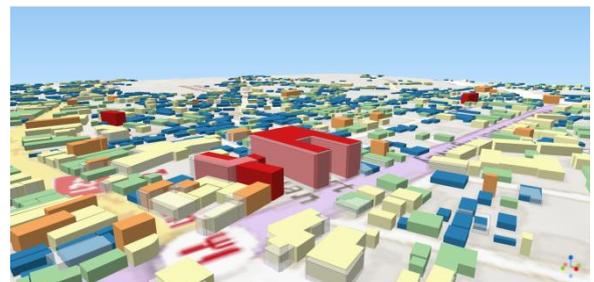


Figure 9. Models showing height and shape difference between OSM buildings and the reference test buildings.

4.2 LOD2 Building Model Generation

A model-driven approach was adopted, which made use of predefined roof models to be fitted on the footprint and height values. A separate footprint for each roof substructure is ideal for roof type identification which can later be followed by roof modelling using the decomposed roof planes. However, there were several limitations in the reconstruction workflow due to the nature of the data used.

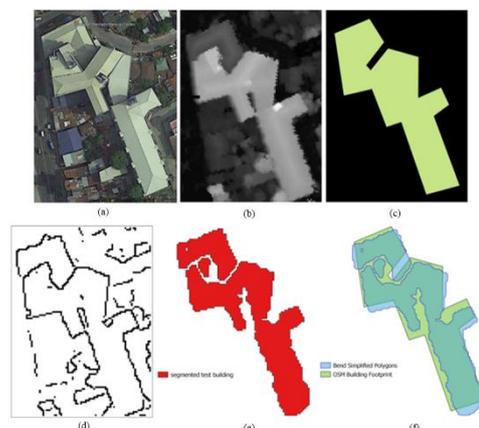


Figure 10. Daniel O. Mercado Medical Center. (a) aerial view taken from Google Earth, (b) LiDAR nDSM, (c) building footprint, (d) edge detection from LiDAR nDSM, (e) segmented

test building from LiDAR nDSM, (f) OSM building footprint overlaid on the refined test building.

As an example, the building shown in Figure 10 has 5 different roof sub structures based on its aerial view. Upon applying methods from Zheng et al. (2017), the result only yielded 3 polygons because nDSM tends to ignore observable visual differences when height values are close to each other, resulting in the lack of segmentation. Additionally, the nDSM-derived polygons exhibit jagged edges even after edge refinement due to it being derived from a raster layer.

Both the OSM and LiDAR data were not sufficient for building segmentation. Introduction of another data source was not considered since the focus of this study is the efficiency and automation of the processes. As a result, an alternative workflow was developed which skipped the building segmentation and proceeded directly with the roof identification. It prioritizes the OSM data due to the straight edges of the building footprints as compared to the jagged edges of the nDSM-derived delineated buildings which requires more advanced processing techniques; hence, it produced more recognizable building models with less intensive computations. However, the roof types obtained are prone to overgeneralization because of several limitations such as (1) the inaccuracy of the OSM traces leading to incorrect analysis of intersecting delineated building polygons, (2) the lack of segmentation resulting to the evaluation of all delineated building polygons within the OSM building footprint, and (3) the medium resolution of LiDAR data over a highly urbanized area.

The slope and aspect information of each pixel from the nDSM were extracted and examined to identify the roof type. The roof planes were then classified based on their orientation values as prescribed by the initial criteria from Zheng et al. (2017). The five roof types were gable, pyramid, flat, complex, and undetected. Undetected roofs can be attributed to the over segmented aspect map wherein roof planes were barely recognizable and small in area with respect to the building footprint as observed in the case of Daniel O. Mercado Medical Center.

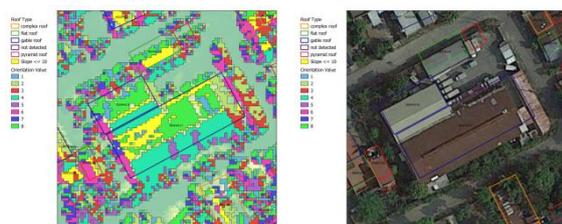


Figure 11. Gable roof building's OSM building footprints overlaid on the aspect and slope map (left) and aerial photo (right).

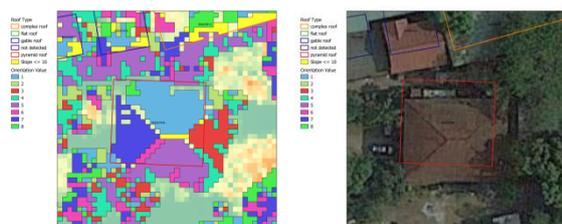


Figure 12. Pyramid roof building's OSM building footprints overlaid on the aspect and slope map (left) and aerial photo (right).

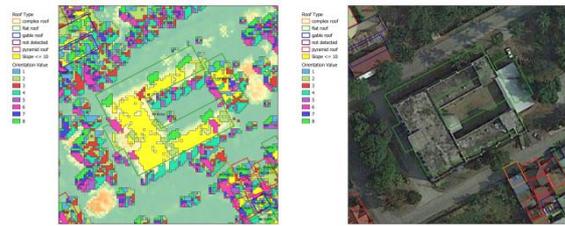


Figure 13. Flat roof building's OSM building footprints overlaid on the aspect and slope map (left) and aerial photo (right).

LOD2 3D buildings were generated using a model-driven approach thus creating building models with generalized roof structures (Figure 14). Multipatch features were utilized for the building reconstruction through a combination of triangle strip and triangle fan part types. The current workflow can only model regularly shaped buildings with basic and symmetric roof types such as gable, pyramid, and flat roofs because more complicated buildings require step edge segmentation and ridge decomposition which is not possible with the data used. Figures 15 and 16 show samples of the different types of reconstructed 3D building models together with their aerial view.

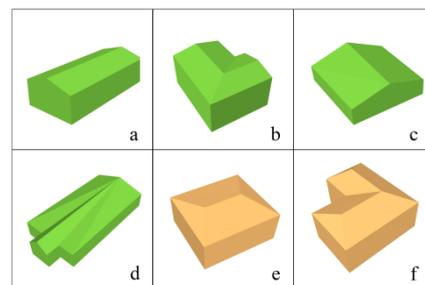


Figure 14. 3D building models for (a) rectangular gable, (b) l-shaped gable, (c) irregular pentagon gable, (d) irregular gable, (e) rectangular hipped, and (f) l-shaped hipped.

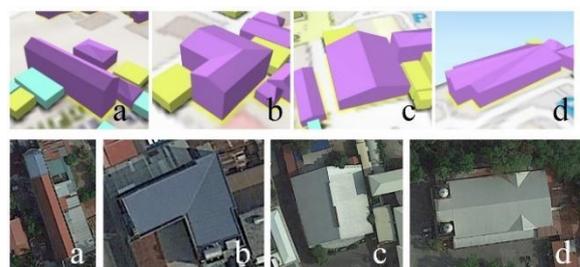


Figure 15. Sample 3D reconstructed building models for (a) rectangular gable, (b) l-shaped gable, (c) irregular pentagon gable, (d) irregular gable.

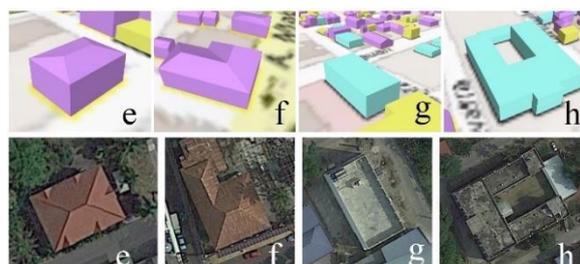


Figure 16. Sample 3D reconstructed building models for (e) rectangular hipped, (f) l-shaped hipped, (g) rectangular flat, (h) irregular flat.



Figure 17. Panoramic view of the LOD2 3D City Model.

The workflow was able to reconstruct 3706 out of 6633 or 55.87% of the buildings in 3D. These LOD2 3D building models were assessed by comparing 462 test buildings to its counterpart as seen from Google Earth in replacement of ground validation procedures. A confusion matrix (Table 5) for the roof type classification was created resulting in an overall accuracy of 44.37%. The roof identification criteria were dependent on the number of roof planes and the aspect values. It was therefore prone to misclassification due to over segmentation of roof planes.

Extracted Data	Reference Data						User's Accuracy
	Gable	Hipped	Complex	Flat	Unclassified	Total	
Gable	48	20	4	3	0	75	64
Hipped	99	97	15	6	0	217	44.70
Complex	25	27	10	8	0	70	14.29
Flat	23	12	7	50	0	92	54.35
Not detected	4	2	2	0	0	8	0
Total	199	158	38	67	0	205	-
Producer's Accuracy	24.1	61.39	26.32	74.63	-	-	44.37%

Table 5. Confusion matrix for roof shape classification.

4.3 Automation in QGIS

The workflow for both the LOD1 and LOD2 3D city model generation were automated using the graphical modeler tool and the python console in QGIS. The two test benches used were a Lenovo Ideapad 320 (Test Bench 1) and a Dell Inspiron 5570 (Test Bench 2) with their specifications shown in Table 6.

	Lenovo Ideapad 320 (Test Bench 1)	Dell Inspiron 5570 (Test Bench 2)
Processor	Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 1.99 GHz	Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz
Installed RAM	8.00 GB	8.00 GB
Storage	240 GB SSD	500 GB SSD
Operating System	Windows 10	Windows 10

Table 6. Laptop Specifications.

The processing time of each tile for both test benches are shown in Tables 7 and 8, respectively. Test bench 1, with a lower processor, was able to finish the entire procedure with a total time of 3 hours, 3 minutes, and 24.6 seconds, whereas test bench 2, with the more advanced processor chip, took 1 hour, 37 minutes, and 15.4 seconds of processing time.

Tile Number	Run Time (H:MM:SS.S)		
	Height Extraction (LOD1)	LOD2	Total
E299N1558	0:03:39.8	0:37:44.9	0:41:24.7
E299N1559	0:02:39.2	0:31:33.8	0:34:12.9
E300N1558	0:03:50.2	1:14:20.0	1:18:10.2
E300N1559	0:02:45.4	0:26:50.4	0:29:35.8
Total	0:12:54.5	2:50:29.1	3:03:24.6

Table 7. Overall Processing Time (Test Bench 1).

Tile Number	Run Time (H:MM:SS.S)		
	Height Extraction (LOD1)	LOD2	Total
E299N1558	0:02:35.5	0:23:14.2	0:25:49.6
E299N1559	0:02:28.2	0:18:05.8	0:20:34.0
E300N1558	0:01:41.3	0:30:08.9	0:31:50.2
E300N1559	0:02:42.2	0:16:19.4	0:19:01.6
Total	0:09:27.2	1:27:48.2	1:37:15.4

Table 8. Overall Processing Time (Test Bench 2).

5. CONCLUSIONS

The LOD1 3D City Model workflow was able to reconstruct 70.66% 3D building models from the 4 LiDAR tiles with a maximum processing time of 12 minutes and 54.5 seconds for test bench 1 and 9 minutes 27.2 seconds for test bench 2. On the other hand, the LOD2 3D City Model workflow was able to reconstruct 55.87% 3D building models with an overall accuracy of 44.37% from the 4 LiDAR tiles. The maximum processing time for test bench 1 was 2 hours 50 minutes and 29.1 seconds while test bench 2 took 1 hour 27 minutes 48.2 seconds. This proved that efficient generation of LOD1 and LOD2 3D City Model from freely available and already existing data can be achieved in the FOSS environment using less computationally intensive GIS-based algorithms.

The OSM building footprints are up to date but inaccurate and have inconsistent standards. It is therefore recommended to obtain building footprints with more accurate tracing and positioning. It would be best to have a standard for tracing such as one footprint for each building, so that the methodology can adapt better to the available data. On the other hand, the LiDAR data is not up to date but is more accurate however it needs to be supplemented by other data with better horizontal accuracy. It is recommended to obtain up-to-date and higher resolution data to better represent high density urban areas. It is also recommended to explore other data sources such as orthophotos for building and edge detection. Validation was done through visual inspection via Google Earth Pro. It is recommended to conduct an onsite ground assessment to verify the accuracy of the building heights and roof models.

Three dimensional digital maps hold the future of smart cities. The resulting model is envisioned as a base map to provide initial

estimation, prediction, and simulation of different applications such as the following: (1) estimation of building heights and levels, (2) valuation of buildings, (3) population estimation, (4) visualization, (5) designing green areas, (6) 3D cadastre, (7) change detection, (8) routing, (9) evacuation planning, and (10) flood application. The sharing and integration of information in a developing country, like the Philippines, shall transform its urban areas to more inclusive, smarter, and more sustainable cities.

ACKNOWLEDGEMENTS

This research would not have been possible without the collaborative efforts of the developers and researchers behind open-source software and open data. The authors would like to express their gratitude to the volunteer mappers for their invaluable contribution in tracing satellite imagery into OpenStreetMap, Geofabrik for providing daily extracts of OpenStreetMap dataset, and UP TCAGP and their DREAM and Phil-LIDAR 1 Programs for the public release of the Taal Open LiDAR data.

REFERENCES

- Arroyo Otori, K., Biljecki, F., Kumar, K., Ledoux, H., Stoter, J. (2018). Modeling Cities and Landscapes in 3D with CityGML. In *Building Information Modeling*, Springer, pp. 199–215. doi: 10.1007/978-3-319-92862-3_11
- Awrangjeb, M., Zhang, C., Fraser, C. S. (2013). Automatic extraction of building roofs using LIDAR data and multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 83, 1-18. doi:10.1016/j.isprsjprs.2013.05.006
- Biljecki, F., Zhao, J., Stoter, J., Ledoux, H. (2013). Revisiting the concept of level of detail in 3D city modelling. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-2/W1: 63–74. doi: 10.5194/isprsannals-II-2-W1-63-2013
- Biljecki, F., Ledoux, H., Stoter, J. (2014). Height references of CityGML LOD1 buildings and their influence on applications. *Proceedings of the 9th 3D GeoInfo Conference 2014*. ScholarBank@NUS Repository. <https://doi.org/10.4233/uuid:09d030b5-67d3-467b-babb-5e5ec10f1b38>.
- Chen, L., Teo, T., Hsieh, C., Rau, J. (2006). Reconstruction of Building Models with Curvilinear Boundaries from Laser Scanner and Aerial Imagery. *Advances in Image and Video Technology Lecture Notes in Computer Science*, 24-33. doi:10.1007/11949534_3
- Dorninger, P., Pfeifer, N. (2008). A Comprehensive Automated 3D Approach for Building Extraction, Reconstruction, and Regularization from Airborne Laser Scanning Point Clouds. *Sensors*, 8(11), 7323-7343. doi:10.3390/s8117323
- Esri. (2008). *The Multipatch Geometry Type: An Esri White Paper* [White Paper]. <https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/multipatch-geometry-type.pdf>
- Fu, C., Shan, J. (2008). 3D Building Reconstruction From Point Primitives: A Geometric Graph Approach. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVII.
- Gröger, G., Kolbe, T.H., Czerwinski, A., Nagel, C. (2008). *OpenGIS City Geography Markup Language (CityGML) Encoding Standard*. Version 1.0.0., Open Geospatial Consortium, OGC Doc. No. 08–007r1.
- Gröger, G., Plümer, L. (2012). CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, 12–33. doi:10.1016/j.isprsjprs.2012.04.004
- Lawhead, J. (n.d.). *PyShp*. PyPI. <https://pypi.org/project/pysph/>.
- Martin, K., Pengson, L., Bernardez, G., Sinnaco, M., Soriano, M., Pascua, C. (2014). Building Footprint Extraction and Tree Removal in Lidar-Derived Digital Elevation Models. *Phil-LiDAR 1 Project*, University of the Philippines Baguio. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.740.2365&rep=rep1&type=pdf>.
- Mendes, T. S., Poz, A. P. (2013). Street Region Detection From Normalized Digital Surface Model And Laser Data Intensity Image. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-3/W22, 197-202. doi:10.5194/isprsarchives-xxxviii-3-w22-197-2011.
- Paringit, E., Uy, F. (2017). *LiDAR Surveys and Flood Mapping of Yabaan River, Quezon City*. University of the Philippines Training Center on Applied Geodesy and Photogrammetry. <https://dream.upd.edu.ph/assets/Publications/LiDAR-Technical-Reports/MIT/LiDAR-Surveys-and-Flood-Mapping-of-Yabaan-River.pdf>
- Sajadian, M., Arefi, H. (2014). A Data Driven Method for Building Reconstruction from LiDAR Point Clouds. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-2/W3, 225-230. doi:10.5194/isprsarchives-xl-2-w3-225-2014
- Kurdi, F. T., Awrangjeb, M., Liew, A. W. (2019). Automated Building Footprint and 3D Building Model Generation from Lidar Point Cloud Data. *2019 Digital Image Computing: Techniques and Applications (DICTA)*. doi:10.1109/dicta47822.2019.8946008
- Villanueva, J., Ang, M., Inocencio, L., Rejuso, M. (2015). Semi-Automated Building Footprint Extraction, Delineation And 3D Visualisation Of The University Of The Philippines Main Campus From LIDAR Data Using GIS-Based Open Source Software. *Free and Open Source Software for Geospatial (FOSS4G) Conference Proceedings*, Vol. 15 , Article 19. doi: <https://doi.org/10.7275/R5KD1W3F>.
- Zhang, S., Han, F., & Bogus, S. M. (2020). Building Footprint and Height Information Extraction from Airborne LiDAR and Aerial Imagery. *Construction Research Congress 2020*. doi:10.1061/9780784482865.035.
- Zheng, Y., Weng, Q., Zheng, Y. (2017). A Hybrid Approach for Three-Dimensional Building Reconstruction in Indianapolis from LiDAR Data. *Remote Sensing*, 9(4), 310. doi:10.3390/rs9040310.