

3D CITYGML BUILDING MODELLING FROM LIDAR POINT CLOUD DATA

Purnima Jayaraj^{1,*}, Anandakumar M Ramiya²

¹ College of Engineering, Guindy, Anna University, Chennai, Tamilnadu, India – purnima1298@gmail.com

² Indian Institute of Space Science and Technology, Trivandrum, Kerala, India – ramiya@iist.ac.in

Commission V, WG V/8

KEY WORDS: LiDAR point cloud, CityGML, LoD1, LoD2, Smart Cities, 3D building model.

ABSTRACT:

With recent government initiatives for smart cities, 3D virtual city models are in demand for managing and monitoring the urban infrastructure. 3D building models forms an important component of 3D virtual city model. LiDAR remote sensing has revolutionized the way the third dimension can be precisely mapped and proved to be an important source of data for 3D models. The model thus generated should be in an open data format to be used across various applications. CityGML is an open data model framework that enables storage and exchange of 3D models which can be used for diversified applications. The main objective of this research is to develop a methodological workflow to create 3D building models in CityGML standard from airborne LiDAR point cloud. Initially building points were isolated from the airborne LiDAR data using point cloud processing algorithms. 3D building models with levels of detail (LoD1 and LoD2) were generated from the building points in CityGML standard using commercial (ArcGISPro) well as open source packages (3dfier, Citygml4j). Results prove that the models developed using open source packages are comparable to that provided by commercial packages.

1. INTRODUCTION

1.1 Introduction

With close to 95% of urban expansion expected in the coming decade across the globe and with countries across the globe embracing concept of smart cities, precise mapping of the manmade and natural resources becomes crucial (Sustainable cities and communities, UNDP,2018). For improved decision making, 3D maps are becoming more prevalent compared to 2D maps. In 3D maps, the urban objects are represented by 3D geometry. Most of these 3D models are pure graphical models and were mostly used for visualisation in the past decade (Biljecki F et al,2015). In order to fully utilise these 3D model to make informed decision, it is necessary to add spatio-semantic details to the 3D models (Gröger G et al, 2012). For example, instead of defining a 3D building as an object, if walls, roofs, doors etc. along with the functional utility of the building is defined, it can be used for various application. If the 3D building models thus defined is to be used for solar energy potential assessment, then the area, slope, aspect of the roof surface defined can act as an input to the solar empirical model to assess the solar potential of the building. This can be used for installing solar panels in the roof. The 3D models defined through such methods can be used in various application domain if a standardised ontology is defined as there exists a tangles relationship between spatial analysis, use cases and application domains (Gröger G et al,2012). CityGML is thus one such open data model frame work that enables storage and exchange of 3D model which can be used for diversified application. (Gröger G et al,2012). It is the official OGC standard for the modelling and exchange of virtual 3D city model. It defines the classes and relations for the most relevant topographic objects in cities. CityGML model can store semantic information and relationship of the various city objects in the model which is important for various applications. CityGML is interoperable data exchange model which can be used across various geo web services.

One of the main features of CityGML model is that it is well defined with five consecutive levels of detail (LoD). The levels are labelled as LoD0 to LoD4, LoD0 being the coarsest level and LoD4 being the level with high detail of information.

LoD0 are simply footprints or roof edge polygons.

LoD1 is a simple building block model without any roof structures.

LoD2 is the building model with well-defined roof structure.

LoD3 is even more detailed level where the buildings have detailed structure like doors, windows, and bays being represented

The highest accuracy level, LoD4, adds information of interiors of the building such as rooms, stairs, and furniture etc. The figure 1 shows the various level of detail in CityGML model.

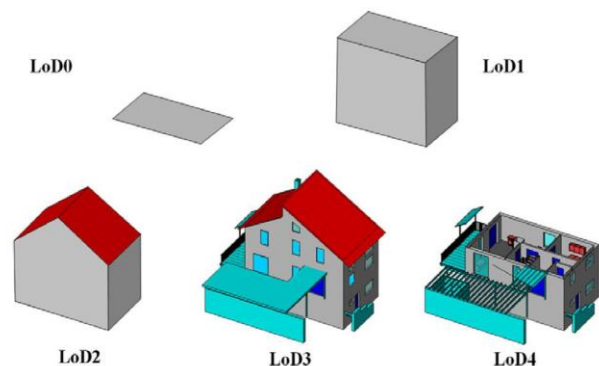


Figure 1 – Representation of different level of details (Gröger & Plümer 2012)

CityGML building models are used in many fields which includes object recognition, navigation, urban studies, traffic planning, finding the optimal location to place a surveillance camera, energy demand estimation, aiding positioning, estimation of the propagation of noise in an urban environment,

* Corresponding author

facility management, forest management, routing, lighting simulations, change detection, estimation of the solar irradiation, in disaster management like flooding where models can be used for analyse the impact of flooding by an overflow of water from water bodies or heavy precipitation to the buildings etc.

Figure 2 shows the various application of CityGML models:

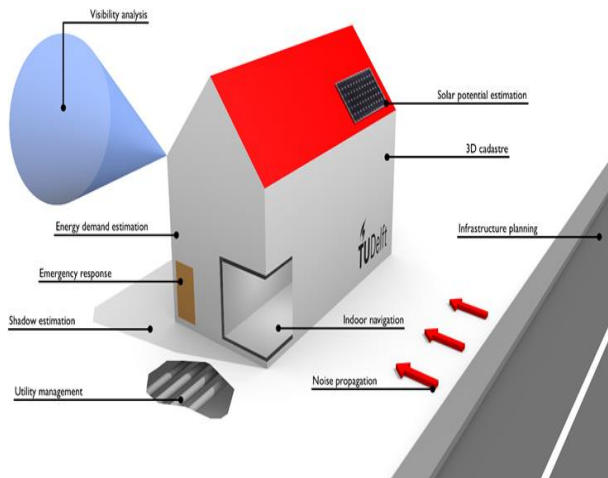


Figure 2 – Various application domain of 3D CityGML models (Biljecki F et al, 2015)

Elevation is one the main components to represent an object in 3D. Elevation from stereo images using photogrammetric techniques is a widely popular geospatial technique (Hirano A. et al,2003). With the advent of LiDAR technology, elevation can be measured at an unprecedented accuracy based on the principle of ranging (Vosselman & Maas,2010; Baltsavias, E., 1999). LiDAR integrates three important technologies – Laser ranging, IMU and GPS to produce georeferenced point cloud (Vosselman & Maas,2010; Baltsavias, E., 1999). The 3D models generated from the LiDAR point cloud are true to geometry. When compared to aerial images, the vertical accuracy from LiDAR is unparalleled. Thus LiDAR point cloud can be used for creating accurate 3D geometry models. There have been several studies which use elevation derived from stereo images and LiDAR for creating CityGML based 3D models. There has been several studies using the stereo image based elevation for creating CityGML based 3D models. When CityGML was recognized as an official standard of Open Geospatial Consortium in 2008, it led many researches to develop a method for modelling the building in CityGML.

Falkowski K et al,2009 employed a semi-automatic approach for generating full CityGML models from images by using IFC-explorer. Automatic detection of features like corners at the top and bottom of buildings or at corners of windows was rather difficult, therefore semi-automatic modelling of buildings was done. Further research involved devising a methodology for automatic modelling of building in CityGML.

With the emergence of more and more very high resolution (VHR) imagery by satellites with ground sampling distances in the range of one meter or below, automatic modelling of the buildings based on CityGML standard (OGC standard-2008) was feasible.

Krauß, T et al, 2008 developed a methodology for automatic modelling of buildings using Ikonos stereo image pairs of the cities of Athens and Munich. After the advent of photogrammetric tools like Erdas imagine LPS, ArcGIS stereo

analyst, many research was carried on using these tools for building CityGML model.

Sengul A in 2012 proposed a methodology in which data (aerial images) plotted by Erdas LPS and stereo analyst for ArcGIS software was transferred to CityGML using Safe Software's Feature Manipulate Engine (FME).

One disadvantage of using the photogrammetric tools such as ERDAS LPS and Stereo Analyst were that the CityGML or complex 3D City models puts high demands on photogrammetric software tools. And it has to cope with 3D geometry types like solid, TINs etc. (Kolbe et al,2009)

Also, Accuracy is an important aspect with 3D city models.

Malamboa, L., and M. Hahn in 2010 described ways of creating multi-level building model using the CityGML model. In this research, Autodesk's LandXplorer Studio was the main software used for creation of CityGML model.

Also, Preka. D & Doulamis. A ,2016 developed a methodology in order to produce a 3D model of 14 buildings that constitute a block at the municipality of Kaisariani, Athens, in Level of Detail 2 using 3 software– SketchUp, PostgreSQL/PostGIS, 3DCityDatabase.

In 2017 Dragana Popovic et al used a series of programs such as Feature manipulation engine, PostGIS to process the collected LiDAR data and orthophoto (with oblique images), transform it into CityGML and store it in spatial database. Mostly commercial software's were used to create 3D CityGML model. However, there has been very limited studies which investigate the potential of open source platform for creating CityGML based 3D models from LiDAR point cloud

Thus the main objective of this research is to create LoD1 and LoD2 3D building models from airborne LiDAR point cloud in CityGML format.

2. METHODOLOGY

2.1 Dataset used

The dataset used for this research is airborne LiDAR point cloud data of Vaihingen an der Enz, Germany. The LiDAR point cloud is projected to WGS 84 / UTM zone 32N coordinate system whose SRID is EPSG:32632. The point cloud data consists of 31415 points which is classified into various class codes including ground, low vegetation, medium vegetation, high vegetation, building etc. with a point spacing and point density of 0.52 m and 4 points per square metre respectively. For this study, only building points were used for creating CityGML building models. Ground points were used to create DTM.

2.2 Creation of LoD1 3D CityGML model

LoD1 3D building model is simple building model without any roof structure. LoD1 3D building model is created using two inputs:

1. LiDAR Point Cloud
2. 2D Building foot print extracted from LiDAR point cloud.

2D building foot print is extracted by 2 methods

2.2.1 Extraction of building foot print from LiDAR point cloud data: The building foot print is extracted by 2 methods as listed below:

1. **By using raster images in ArcGISPro using building footprint extraction tool:**

Extract building foot print from LiDAR toolbox is used to extract building foot print from LiDAR data. This process involves the creation of las dataset of the point cloud and mainly 2 raster tiles-DTM and DSM. After the DTM, DSM tiles are created, composite mosaic dataset is created from them. Finally, Extract draft building footprint tool is used to extract the foot print from the composite dataset.

2. By LiDAR toolbox - LAsTools

LAsTools is an open source tool box for handling .las files. LiDAR point cloud is split into a las file with building points alone using LASSplit. Then, LASBoundary is used for extracting building foot print. LAS boundary reads LiDAR data and computes a boundary polygon for the points. LiDAR point cloud containing building points alone is given as an input to LAS Boundary and Concavity is fixed to 2 and disjoint option is checked. Thus output is a polygon with the building footprint in .shp format.

2.2.2 Creation of LoD 1 CityGML model using 3Dfier:

After the building foot print is extracted by 2 methods, these building foot print along with the LiDAR point cloud is used to create LoD1 3D building model using open source software 3Dfier (Tudelft/3dfier,2018). 3Dfier takes building footprint and LiDAR point cloud as input and creates LoD1 3D model. It uses yaml configuration file to create LoD1 model. The yaml configuration file is modified and LoD1 3D model is created in CityGML by entering a simple command-

```
3dfier name_of_configfile.yml --CityGML  
arg
```

2.3 Creation of LoD2 3D CityGML building model

LoD1 3D CityGML building model created has no detailing of the roof structure of the buildings. For diverse applications like solar energy potential or flood modelling, 3D CityGML model must have detailed roof structure of the buildings. Thus, LoD 2 CityGML model has well defined roof structure. LoD2 3D CityGML building model is created by 2 methods-

1. By commercial software – ArcGISPro Schematic local government scene solution
2. By open source Java API – citygml4j

2.3.1 Creation of LoD2 3D CityGML building model by ArcGISPro- The Schematic Local Government Scene of ArcGIS Pro is a LoD2 (Level of Detail) 3D scene that includes the World Topographic Map draped over detailed elevation along with LoD2 buildings, building floors, and trees. Creation of LoD 2 3D building model involves various steps explained below.

2.3.1.1 Creation of elevation layers from LiDAR data and roof form extraction: Firstly, Las Dataset of the LiDAR point cloud is created. Then, 3D raster layers- DTM, DSM, nDSM is created from las dataset. Roof form need to be extracted in order to create LoD 2 building. Extract roof form tool is used to extract the roof form of the buildings. It uses LiDAR derived raster's such as DTM, DSM, nDSM and vector building footprints extracted from LiDAR point cloud to find both flat and sloped planar areas within a roof area of a building. The tool then estimates a standard architectural form for the roof based on the attributes collected from these planar surfaces. These attributes are then used to perform a procedural rule that generates the roof

features in 3D. This tool calculates or identifies the roof type of the building -flat or gable or hip shown in figure 3

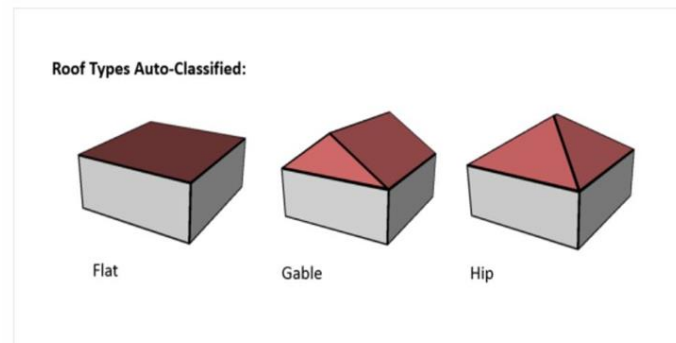


Figure 3 - Different roof forms of building (ArcGIS Pro Roof form extraction process)

Eave height is another attribute which the tool calculates. It is calculated to be the lowest elevation values of the DSM of the largest slope plane to be detected except for flat roofs. For gable, hip, shed roofs, Roof dir. attribute is calculated. It gives the compass direction that the shed roof slope faces and direction that gable roof ridge faces. Parameters such as minimum flat roof area, minimum slope area, minimum roof height is set to 23, 7, 2.5 square meter respectively in order to extract roof form. Thus finally an output is created containing building with different roof forms by using various raster layers and setting some parameters.

2.3.1.2 Creation of LoD2 3D building feature class(3DCIM) using city engine rules: The roof form extracted is reviewed by calculating the root mean square error. RMSE is calculated by confidence measurement tool and applying suitable symbology layer. The las dataset is compared with the roof form and RMSE errors are corrected by modifying the roof form attributes or roof form vertices. The corrected roof form layer is then used to create 3D building feature class (3DCIM) using feature from City Engine rules tool. Rule package used here is LOD2BuildingShells_Meters.rpk. Thus 3D building feature class containing roof forms of building is created as an output.

2.3.1.3 Conversion of LoD2 3D building feature class to LoD2 CityGML building model- CityGML and 3DCIM feature class are complementary. Therefore, interoperability between CityGML and 3DCIM is essential to achieve all the solution capabilities of the 3D Cities. Thus LoD2 3D building feature class is exported to CityGML using 3DCIM_CityGML.tbx tool box. Input is the ESRI geodatabase containing the 3D building feature class. CityGML version 2.0 is chosen. Also, export is done with building foot print for CityGML 2.0 version. Thus, 3D building feature class obtained is exported to create LoD2 CityGML building model.

2.3.2 Creation of LoD2 CityGML building model using open source – citygml4j- Citygml4j is an open source java library that works with OGC City Geography Markup Language (CityGML). It consists of classes for different CityGML models such as building, city furniture, landuse, relief, vegetation, waterbody, tunnel etc. It takes the coordinates of the building as input for generation of LoD2 CityGML model. The methodology consists of various steps mentioned below

2.3.2.1 Extraction of building coordinates from LiDAR data- Coordinates of various buildings is extracted by various

tools of LAsTools. Firstly, height is modified to above ground value from UTM elevation by LAS height tool of LAsTools. So Las file is created with modified z values. Using LASCLIP tool, las file containing individual building points is created. Finally, using LASBoundary, coordinates of the individual building's is extracted in .txt format. Coordinate of the building can be extracted manually by creating DSM of the modified z point cloud and converting it from latitude, longitude to UTM coordinates (x, y).

2.3.2.2 Creation of LoD2 CityGML building by citygml4j code- Coordinates extracted is given as an input to the java code in the form of ground, wall, roof surface for creation of the LoD2 CityGML building. The coordinates for roof surface such as gable roof form is obtained by visualizing LiDAR point cloud and coordinates of roof form is manually extracted. Thus, output is LoD2 3D CityGML building.

3. RESULTS

3.1 Building foot print extraction

Building foot print was extracted successfully by both the methods. Raster tiles- DTM, DSM, composite mosaic dataset was created and building foot print was extracted from raster tiles. LASSplit of LAsTools identified 7013 points as building out of 31415 points. From the split Las file, building foot print was extracted by LASBoundary successfully.

3.2 LoD1 3D CityGML building model creation

LoD1 3D CityGML building model was created by 3Dfier software successfully. Two LoD1 3D CityGML model was created by using LiDAR point cloud and building foot print extracted by raster images and LAsTools. The LoD1 CityGML building model created by using building foot print extracted by raster images is illustrated in figure 4 and LoD1 CityGML building model created by using foot print extracted by LAsTools is illustrated in figure 5.

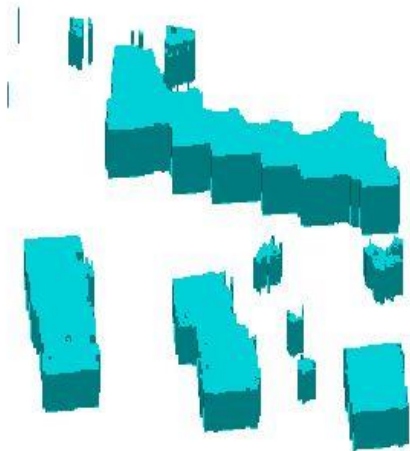


Figure 4- LoD1 3D CityGML model created by LiDAR data and building foot print extracted by raster images

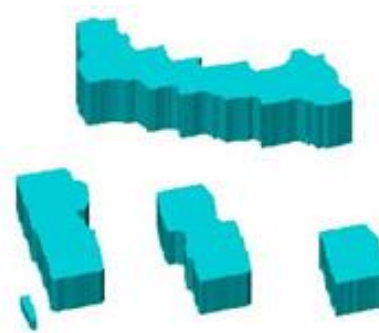


Figure 5- LoD1 3D CityGML model created by LiDAR data and building foot print extracted by LAsTools

3.3 LoD2 CityGML building model creation

3.3.1 Experimental results of ArcGIS Pro- Methodology to create LoD2 CityGML model worked out well using the schematic government solution of ArcGIS Pro. DTM, DSM, nDSM rasters were created and Roof form was extracted using extract roof form tool. LoD2 3D building was created using the rule package of City engine. The 3D building created was exported to CityGML using the CityGML tool box. The LoD2 CityGML model created by ArcGIS Pro is illustrated in fig. 6 below:

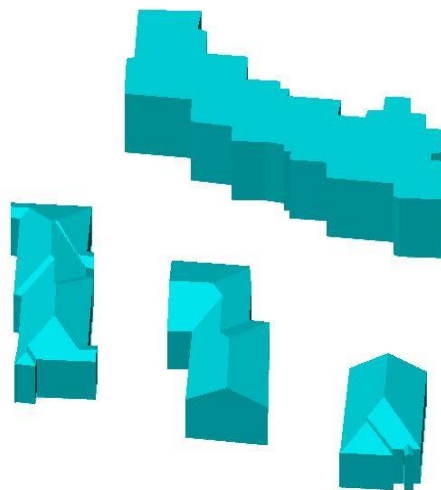


Figure 6-LoD2 3D CityGML model by ArcGIS Pro

3.3.2 Experimental Results of open source java API - citygml4j - Firstly individual building coordinates in form of .txt file was extracted using LASBoundary, LASClip, LASheight tools of LAsTools. Coordinates were given as input in the form of ground, wall, roof surface. The code was run successfully and it resulted in LoD2 CityGML building. The LoD2 CityGML model of individual building is illustrated in figure 7 below:

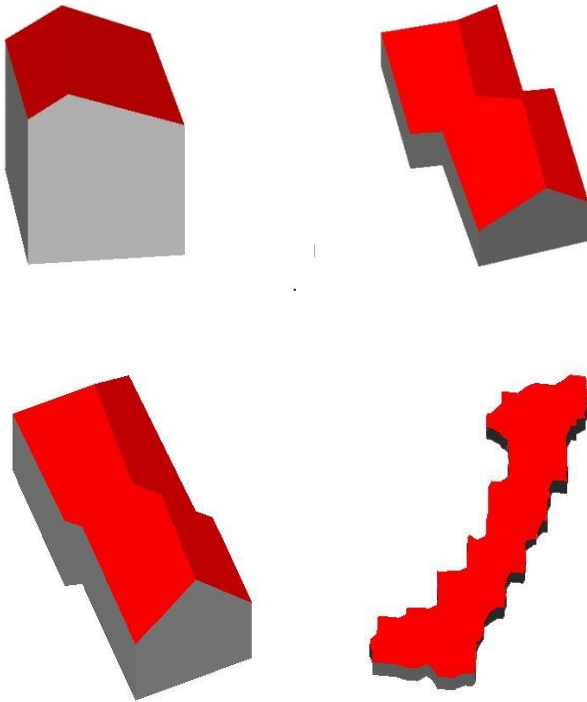


Figure 7- LoD2 CityGML model created by citygml4j

4. DISCUSSIONS

4.1 Discussion

The dataset used for this research is airborne LiDAR point cloud data of Vaihingen an der Enz, Germany. The foot print generated by using raster images could detect the small elevation changes like sharp edges of the building, while the foot print generated by LAStools did not detect the sharp edges. LoD2 CityGML model created by ArcGIS pro was more accurate than created by open source software. The sharp edges of the building were detected by the commercial software whereas the open source code could not detect the sharp edges of the building structure. LoD2 CityGML model created by open source software contains ground, wall, roof surface which can be viewed separately whereas the LoD2 CityGML model by commercial software did not contain any of the surfaces. Export of LoD2 CityGML model created by ArcGIS Pro to kml file to view in the google earth by 3D city database importer/exporter tool using 3d city database, (Yao, Z. at al,2018), created LoD 0 (foot print) model whereas the export of LoD2 CityGML model by open source citygml4j in kml format created LoD1 CityGML model rather than LoD2 CityGML model. Fig 8 shows google earth view created by export of LoD2 CityGML model created by commercial software and Figure 9 shows the export of LoD2 CityGML model created by open source.

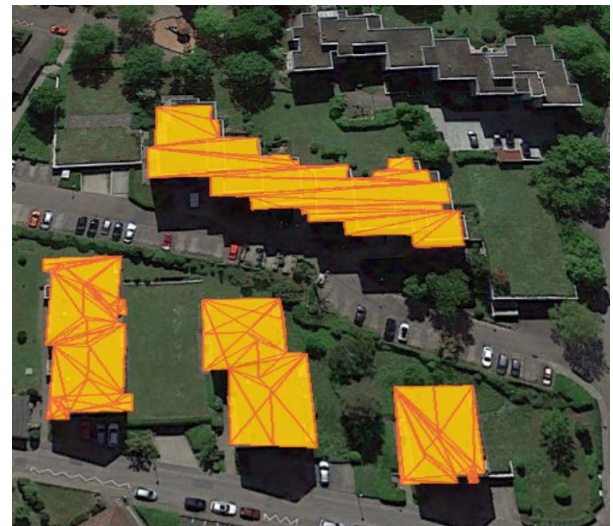


Figure 8- google earth view of LoD2 CityGML model created by commercial software



Figure 9- google earth view of LoD2 CityGML model created by open source software

5. CONCLUSION

5.1 Conclusion

Thus, CityGML building model of different level of details (LoD1 and LoD 2) was created from airborne LiDAR point cloud. LoD1 CityGML model was created using LiDAR point cloud and building foot print which was extracted by two methods in 3dfier software. LoD2 CityGML model was created using both commercial software (ArcGISPro) and open source software(citygml4j). The results prove that the model developed using open source and commercial software are quite reliable and comparable.

5.2 Future research

LoD2 CityGML model created by ArcGIS pro did not contain ground, wall, roof surface separately like that created by open source. Further research needs to be done to export the 3D feature class building to CityGML in form of ground, wall, roof surfaces. Process to create LoD2 CityGML model by open source java API citygml4j was a tedious process. The Code could generate LoD2

buildings individually but not for several buildings at the same time. Hence, for creating building model of more than 100 building, entering the coordinates for each of the 100 building in form of ground, wall, roof surface will require more time and will become a tiresome process. Thus a methodology must be developed to automate the process of creation of LoD2 CityGML model by citygml4j. The code to build the LoD2 CityGML building must be able to read the point cloud data in .txt/.las file format and create LoD2 building model automatically by reading the coordinates in form of ground, wall, roof surface. Also, the conversion of 3D building feature class to CityGML using importer exporter tool in ArcGIS pro resulted in many errors and the result obtained after exporting was LoD2 CityGML building model with LoD 0 geometry only. So when this was exported to kml file to view in Google Earth using 3dcity database importer exporter tool it could be exported as LoD 0-foot print only and not as LoD 1 or LoD 2 (figure 8). On the other hand, export of LoD2 CityGML model by open source software into kml file resulted in LoD1 building model not as LoD2 building model (fig 9). Thus, further research needs to be done to export LoD2 CityGML model created by both commercial and open source software into kml file with LoD2 geometry to view it on google earth similar to that of open street map.

REFERENCES

- Baltsavias E.P, Airborne laser scanning: basic relations and formulas, ISPRS Journal of Photogrammetry and Remote Sensing, vol.54, issue.2-3, pp.199-214, 1999. DOI : 10.1016/S0924-2716(99)00015-5
- Biljecki, F.; Stoter, J.; Ledoux, H.; Zlatanova, S.; Çöltekin, A. Applications of 3D City Models: State of the Art Review. ISPRS Int. J. Geo-Inf. 2015, 4, 2842-2889.
- Citygml4j, the open source java API for CityGML, Retrieved 23 June,2018 from <https://github.com/citygml4j/citygml4j>
- CityGML, OGC Making Location count. Retrieved 28 June,2018 from <https://www.citygml.org/>
- Dragana Popovic , Miro Govedarica, Dusan Jovanovic, Aleksandra Radulovic, Vlado Simeunovic, (2017). 3D Visualization of Urban Area Using Lidar Technology and CityGML 3D Visualization of Urban Area Using Lidar Technology and CityGML. <https://doi.org/10.1088/1755-1315/95/4/042006>
- Falkowski, K., Ebert, J., Decker, P., Wirtz, S., & Paulus, D. (2009). Semi-automatic generation of full CityGML models from images. *Geoinformatik 2009*, 1–3. Retrieved from http://www.geoinformatik2009.de/public/abstracts/abstract_falkowski_ebert_decker_wirtz_paulus.pdf
- Gröger, G., Kolbe T. H., Nagel C., et al., 2012. OGC City Geography Markup Language (CityGML) Encoding Standard, Version 2.0.0. Open Geospatial Consortium Inc.(OGC Doc. No. 12–019).
- Gröger, G. & Plümer, L. 2012. CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*. Vol. 71. pp 12-33. <https://doi.org/10.1016/j.isprsjprs.2012.04.004>
- Hirano, A., Welch, R., & Lang, H. (2003). Mapping from ASTER stereo image data: DEM validation and accuracy assessment. *ISPRS Journal of Photogrammetry and Remote Sensing*, 57(5–6), 356–370. [https://doi.org/10.1016/S0924-2716\(02\)00164-8](https://doi.org/10.1016/S0924-2716(02)00164-8)
- Kolbe, T. H., Nagel, C., & Stadler, A. (2009). CityGML – OGC Standard for Photogrammetry ? Processing, 265–277. Retrieved from <http://www.ifp.uni-stuttgart.de/publications/phowo09/270Kolbe.pdf>
- Krauß, T., Lehner, M., & Reinartz, P. (2008). Generation of coarse 3D models of urban areas from high resolution stereo satellite images. *International Archives of Photogrammetry and Remote Sensing*, 37(B1), 1091–1098.
- Malamboa, L., and M. Hahn. "Lidar assisted citygml creation." *AGSE 2010* (2010): 13
- M.Isenburg, "LAStools - efficient LiDAR processing software" (version 141017, unlicensed), obtained from <http://rapidlasso.com/LAStools>.
- Preka, D., & Doulamis, A. (2016). 3D building modeling in LoD2 using the CityGML standard. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(2W2), 11–16. <https://doi.org/10.5194/isprs-archives-XLII-2-W2-11-2016>
- Roof form extraction process, Retrieved 23 June,2018 from <http://solutions.arcgis.com/local-government/help/local-government-scenes/get-started/roof-form-extraction/>
- Sengul, A. (2012). Extracting Semantic Building Models From Aerial Stereo Images and Conversion To Citygml. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIX-B3(September), 321–324. <https://doi.org/10.5194/isprsarchives-XXXIX-B3-321-2012>
- Sustainable cities and communities, UNDP, Retrieved on 5 Oct 2018 from www.undp.org/content/undp/en/home/sustainable-development-goals/goal-11-sustainable-cities-and-communities.html
- tudelft3d/3dfier, Retrieved on 1 July 2018 from <https://github.com/tudelft3d/3dfier>
- Vosselman, G. & Maas, H-G. 2010. Airborne and terrestrial laser scanning. Whittles Publishing, Scotland. p. 318. ISBN 978-1904445-87-6.
- Yao, Z., Nagel, C., Kunde, F., Hudra, G., Willkomm, P., Donaubaue, A., ... Kolbe, T. H. (2018). 3DCityDB - a 3D geodatabase solution for the management , analysis , and visualization of semantic 3D city models based on CityGML. *Open Geospatial Data, Software and Standards*, 3(1), 5. <https://doi.org/10.1186/s40965-018-00>