Towards a Digital Twin of Liege: The Core 3D Model based on Semantic Segmentation and Automated Modeling of LiDAR Point Clouds

Zouhair Ballouch [1,2,] *, Imane Jeddoub [1], Rafika Hajji [2], Jean-Paul Kasprzyk[1] and Roland Billen [1]

[1] GeoScITY, UR SPHERES, University of Liège, 4000 Liège, Belgium; zouhair.ballouch@student.uliege.be; i.jeddoub@uliege.be; jp.kasprzyk@uliege.be; rbillen@uliege.be

[2] College of Geomatic Sciences and Surveying Engineering, Hassan II Institute of Agronomy and Veterinary Medicine, Rabat 10101, Morocco; r.hajji@iav.ac.ma

*Correspondence: zouhair.ballouch@student.uliege.be

KEY WORDS: Digital Twin, 3D city model, Semantic Segmentation, LiDAR point cloud, CityJSON.

ABSTRACT:

The emergence of Digital Twins in city planning and management marks a contemporary trend, elevating the realm of 3D modeling and simulation for cities. In this context, the use of semantic point clouds to generate 3D city models for Digital Twins proves instrumental in addressing this evolving need. This article introduces a processing pipeline for the automatic modeling of buildings, roads, and vegetation based on the semantic segmentation results of 3D LiDAR point clouds. It employs a semantic segmentation approach that integrates multiple training datasets to achieve precise extraction of target objects. Open-source reconstruction tools have been adapted for building and road modeling, while a Python code was optimized for tree modeling, leveraging a foundational code. The case study was conducted in the city of Liège, Belgium. The obtained results were satisfactory, and the schemas and geometry of the developed models were validated. An evaluation of the adopted reconstruction methods was conducted, along with their comparison to other methods from the literature.

1. INTRODUCTION

Digital Twins (DTs) for cities have become an efficient and collaborative decision-making tool that helps overcome cities' challenges (Ketzler et al. 2020; Jeddoub et al. 2023). The Urban Digital Twins (UDTs) serve city needs by integrating data, models, and processes into a one-stop platform, enabling two-way flows from the physical world to the digital replica and vice versa (Lehtola et al. 2022). As we embark on the journey of implementing UDTs, semantic 3D city models gain perspective (Stoter, Arroyo Ohori, et Noardo 2021). Over the past decades, many scholars have increasingly focused on the creation and use of 3D city models beyond simple visualization (Stoter et al. 2020). Indeed, semantic 3D city models offered many potential applications to urban and geospatial analysis and application at the city scale based on open standards such as CityGML[1] (Biljecki et al. 2015).

Up to date, many city models are spread worldwide, implemented using different data and approaches, and serve various purposes. In contrast, there is a lack of a standard or common framework for 3D city modeling, which was one of the main motivations behind the work of (Lei, Stouffs, et Biljecki 2022). The authors designed a holistic instrument to benchmark and evaluate 3D city models worldwide. Based on the findings, cities, such as Brussels and Namur, have invested in the creation of their 3D city models in the Belgian context.

Inspired by current digital technologies and recognizing the relevance of UDTs in the context of urban planning and management, the city of Liege has invested in the implementation of 3D city models as the first step toward the development of DTs. This study presents the results of SEM3D (3D semantic object extraction for urban application), a project supported by Digital Wallonia and conducted in the GeoScITY[2] lab with the collaboration of the city of Liège. The main contribution is to automatically extract 3D semantic objects for urban applications and explore the 3D modeling process to create 3D models of the derived urban objects. The paper proposes and tests an overall framework, from data preparation to 3D modeling. The particularity of this approach is that it is not restricted to a specific urban object (e.g., buildings) but also enables the modeling of other thematic objects (i.e., roads and vegetation) using open-source tools and the semantic segmentation results of 3D LiDAR data. The main contribution is to shed light on the relevance of the semantic segmentation of 3D airborne LiDAR data in the city modeling framework. The proposed framework uses the existing data and adapts available open-source tools to create a standardized CityJSON[3] 3D city model of common urban objects. The paper is structured as follows: Section 2 gives an overview of the use of semantically segmented point cloud data in city modeling processes. Section 3 presents the proposed workflow, ranging from data preprocessing and transformation to 3D modeling. A description of the study area is provided in the same section. Section 4 discusses the findings. Section 5 concludes and gives an outlook for future perspectives.

2. RELATED WORKS

Digital twins for cities are data-hungry platforms (Batty 2018; Masoumi et al. 2023). They are based on heterogeneous data sets (geospatial data and sensor data, to name a few). 3D point cloud data from airborne acquisition is the most common input data for DTs' implementation. They have shown their

[1] https://www.ogc.org/standard/CityGML/

[2] https://www.geoscity.uliege.be/cms/c_12409035/en/geoscity

[3] https://www.cityjson.org/

capabilities as an input layer for city modeling, namely for 3D building modeling (Jeddoub et al. 2024). For instance, 3D Registration of Buildings and Addresses (3D BAG[4]) is 3D reference building data covering the whole Netherlands in several output formats. The data are provided based on an automatic 3D reconstruction pipeline at different levels of detail (LoD). The workflow uses the 2D building footprints and the AHN point cloud data (the national height model of the Netherlands) acquired by airborne laser scanning (ALS). Furthermore, 3dfier[5] is an automatic workflow and open-source software that reconstructs LoD1 models using classified LiDAR point cloud data in LAS format and 2D semantic polygons (i.e., building footprints, water bodies, etc.) (Ledoux et al. 2021). The workflow is based on a set of rules and uses a configuration file to generate the 3D model. The software has support for various formats. In addition, the authors in (Nys, Billen, et Poux 2020) have proposed an automatic CityJSON workflow that extracts roof surfaces from LiDAR data and generates LoD2.1 building models. Another related work, City3D, was conducted by (Huang et al. 2022), presenting a large-scale 3D building reconstruction from the ALS point cloud. The authors propose an approach that infers the vertical walls of buildings from airborne LiDAR point clouds. In their work, the authors address in a comprehensive way the challenges related to large-scale urban reconstruction from ALS data, namely: building instance segmentation, incomplete data, and complex structures. However, many of these processes do not involve advanced classification of the point cloud data. In this regard, enhancing semantic segmentation-based Artificial Intelligence (AI) approaches improves the use of 3D point cloud data, thus creating 3D urban models (Ballouch et al. 2022; 2024). It enables the automatic extraction of single and multiple city objects, which simplifies object modeling. Many DT initiatives have acquired point cloud data (airbone or unmanned aerial vehicle data) to create and enrich their semantic 3D city models (DImitrov et Petrova-Antonova 2021; Peters et al. 2022; Khawte et al. 2022). In fact, optimizing the semantic segmentation process is of great interest to reconstruct 3D city models and implement UDTs efficiently and correctly.

3. MATERIALS AND METHODS

This section explains the methodology used in the framework of this work. It consists of three main steps: data collection, data processing, and 3D modeling. The workflow is summarized in Figure 1. The pink boxes describe input data (namely point cloud data, topographical data, and images), while the blue ones refer to the intermediate transformations and processes. The resulting CityJSON 3D city model is presented in a green box. In the following, we first describe the input data. Then, we explain the semantic segmentation process and, finally, the reconstruction process for each city object.
The first step involves the collection of the input data sets. Then, the second step takes raw point cloud data and performs a semantic segmentation process. For this, an AI-based approach is used. The approach fuses ALS point clouds with corresponding aerial photos. It can accurately extract the main 3D objects within an urban scene with both geometric precision and semantic richness. Deploying a fusion approach with other sources (aerial photos, satellite images, etc.) allows for combining the spectral richness of images and the altimetric accuracy of 3D point clouds. Our aim is to automate the extraction of 3D objects, such as roads, vegetation, etc., in our study area, presented subsequently in section 3.2, with high accuracy and performance.

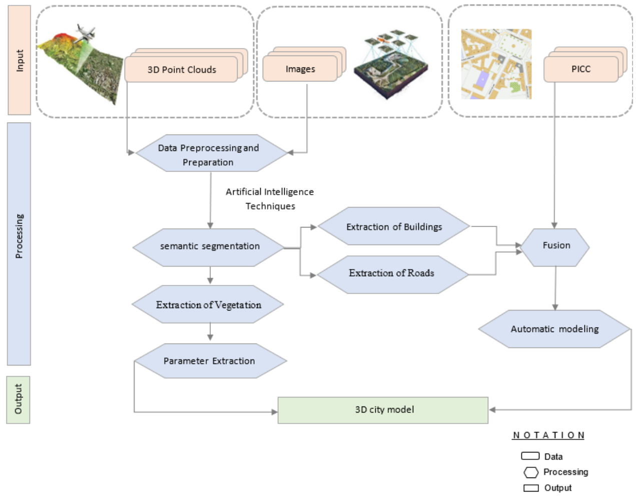

Figure 1. The general workflow.

The third step is dedicated to the modeling process. For each urban object, an approach or an open-source tool is deployed. The extracted semantic classes from the semantic segmentation of 3D point cloud data are assigned to each modeling pipeline. For instance, to model buildings using open standards (i.e., CityJSON), GeoFlow[6] uses the building point cloud data as well as the building footprints to automatically generate the 3D building models at LoD2. For the road modeling, the class number of the corresponding road point cloud data is specified in the configuration file necessary to run the open-source tool 3dfier. The same logic is applied to the vegetation modeling. The derived vegetation point cloud data were integrated into the modeling process based on an adapted code. This code[7] was based on the fundamental code previously available as open source.

3.1 Data collection

The data sources include LiDAR point clouds and PICC[8] (Plan d'Information sur le Cadre de Cartographie) data. The PICC serves as the three-dimensional digital cartographic reference for the entire Wallonia region in Belgium, with precision less than 25 cm, comprehensively capturing all identifiable elements of the Walloon landscape, such as buildings, structures, railway networks, hydrographic networks, roadways (including lanes, edges, sidewalks, etc.), and more. The datasets were provided by the Walloon region in Belgium. Additionally, other datasets, namely the SUM-Helsinki dataset and the SensatUrban dataset, were acquired through free downloads via links provided later (refer to Table 1). Consequently, the Liège dataset was created by us based on the region's data. Table 1 provides a description of the data sources.

[4] https://3dbag.nl/en/viewer
[5] https://tudelft3d.github.io/3dfier/
[6] https://github.com/geoflow3d/geoflow-bundle
[7] https://github.com/RobbieG91/TreeConstruction
[8] https://geoportail.wallonie.be/georeferentiel/PICC

	Type of object	File type	Geometry/ Data type	Num. objects	Description
Semantic segmentation data	SUM-Helsinki dataset	.PLY	Mesh	19 M	This dataset covers approximately 4 km2 in Helsinki, Finland, featuring six classes: Terrain, Vegetation, Building, Water, Vehicle, and Boat. Derived from 2017's 3D textured meshes of Helsinki with a ground sampling distance of 7.5 cm, the dataset is obtained through oblique aerial images and processed using ContextCapture software. The study area is concentrated on the central region of Helsinki, encompassing 64 selected tiles.
	SensatUrban dataset	.PLY	Point	2847.1M	The SensatUrban dataset collected by UAV over Birmingham, Cambridge, and York cities covers six square kilometers of urban area and features 13 semantic classes with 6 attributes per point: X, Y, Z, and RGB information
	Liège dataset	.LAS	Point	25.635.237	The Liège dataset is derived from the data of the Walloon region described in section 3.1 and includes 12 semantic classes ('Ground', 'High Vegetation', 'Buildings', 'Walls', 'Bridge', 'Parking', 'Rail', 'Traffic Roads', 'Street Furniture', 'Cars', 'Footpath', 'Bikes', 'Water'). Each data point within the dataset is characterized by 6 attributes: X, Y, Z, and RGB information.
Modeling data	point clouds from the "Outremeuse" neighborhood in Liège	.LAS	Point	10.619.980	It is a point cloud of the Outremeuse neighborhood in Liège, extracted from four point cloud tiles covering this area, using vector data representing the administrative boundaries of the city of Liège
	PICC (building surfaces, road axes, road edges	.SHP	Building surfaces: Polygon	Building footprints :3897	These are vector data from the Walloon region available upon request. Each object (building,road axe,,etc) is represented by an identifier and attribute information relative to it
			Road axes: Ligne	Road axis:342	
			Road edges: Ligne	Road edges:1607	

Table 1. Data sources.

3.2 Semantic segmentation of 3D LiDAR points clouds

The quality of semantic segmentation results plays a crucial role in the geometric accuracy of 3D urban models created based on these results. Therefore, the choice of a semantic segmentation approach that accurately extracts urban objects is essential. To achieve this, the LiDAR point clouds were fused with their corresponding images using the "RandLaNet" deep learning model (Hu et al. 2020). This model was adopted for semantic segmentation due to its documented performance in the literature (Hu et al. 2020; Guo et al. 2021). In this study, we trained this model on three different datasets: SensatUrban, accessible at https://github.com/QingyongHu/SensatUrban; a dataset from https://3d.bk.tudelft.nl/projects/meshannotation/; and a dataset created in the urban context of Liège city (access to this data is available upon request). Model parameters and hyperparameters were adjusted. The predictions made by the trained model were based on point cloud data from the "Outremeuse" neighborhood in Liège, Belgium. The location of this neighborhood is shown in Figure 2 below. The data used for creating the Liège dataset and working on the Outremeuse neighborhood are from recent LIDAR acquisitions in the Walloon region of Belgium (2021–2022).
The data characteristics include an average flight altitude (AGL) of 2400 m, density of 6.8 points/m², and the use of Double LMSQ780 and Double VQ780II-S equipment. The data were provided in 8 blocks in". LAS" format, adhering to ETRS89 / Lambert Belgian 2008 planimetric coordinates, Second General Leveling altimetric coordinates, and planimetric accuracy with RMSE <= 1 m and altimetric accuracy with RMSE <= 0.4 m. A preprocessing step, including cleaning, was conducted to ensure data consistency. After adjusting projections and merging LiDAR point clouds with corresponding images (see Figure 2), the Outremeuse neighborhood data were prepared for prediction, while data from other areas in Liège were utilized for creating the third training dataset. Data preprocessing was performed using CloudCompare, and data preparation and processing were carried out using the Ubuntu tool.

The model "RandLaNet" has already been validated through our previous studies as well as by several studies in the literature using various evaluation criteria, including measures such as Accuracy, Intersection over Union (IoU), Recall, F1-score, and Confusion matrix (Hu et al. 2020; Guo et al. 2021; Ballouch et al. 2022). Therefore, in this study, we have opted just for visual validation through a comparison of the model's results with the ground truth (see Figure 3), considering the comprehensive set of evaluation metrics used in prior research.

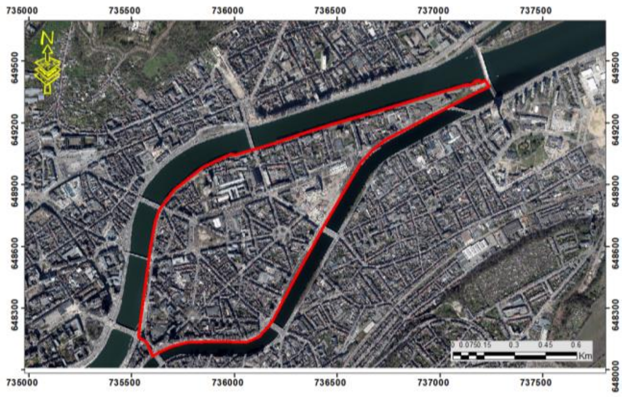

Figure 2. Geographical location of the Outremeuse district.

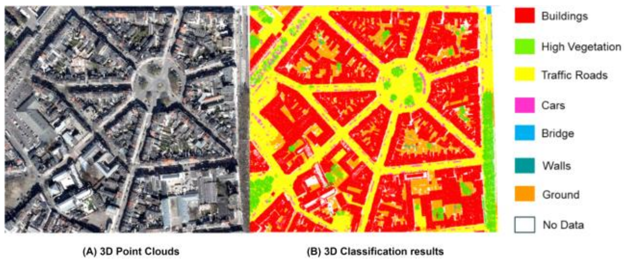

Figure 3. (A) 3D point cloud representation and (B) example of 3D semantic segmentation outputs- Outremeuse district.

3.3 3D modeling workflow

This section outlines the processing pipeline we followed for generating 3D models from a classified point cloud. The required data include the classified point cloud obtained from section 3.2 and the PICC.

3.3.1 Automatic 3D buildings modeling

Building modeling was conducted using the GeoFlow tool, an open-source tool for 3D building model reconstruction from point clouds. The objective is to generate a realistic three-dimensional representation of buildings by harnessing point cloud data, vector data (PICC), and modeling functions provided by GeoFlow.

To execute the reconstruction from input data, both a JSON file containing a flowchart describing the logic of the reconstruction and the executable GeoFlow are necessary components. The flowchart outlines how different plugins and nodes connect, while GeoFlow executes the logic defined in the flowchart.

3.3.2 Automatic roads modeling

Studies and methods for 3D road modeling are still limited. The focus was historically on 3D building models. This is due to the lack of complete data and because most 3D roads have linear representation. However, a recent study conducted by (Yarroudh, Nys, et Hajji 2023), has proposed an automatic process of 3D road modeling based on CityGML 3.0 specifications and CityJSON encoding. The authors produce a LoD2 3D road model using a semi-automatic extraction workflow based on mobile mapping LiDAR data.

Given the type of data provided in the scope of this work, we opted for the two approaches below: the first approach relies on developing an FME workbench. FME is an ETL (Extract, Transform, and Load) process that allows a series of data transformers. It also has the capability to read, convert, and write many data types and formats. Initially, we created an FME workspace (refer to

Appendix) for the roads using only the 2D road axis and the georeferenced DEM generated from the LiDAR data using the "SurfaceModeller" transformer. We create a CityJSON v1.0.1-compliant model describing the road. The FME workbench is reusable. For now, the workbench allows the generation of the LoD1 road model. Further work will help extend it to produce higher LoDs. The second approach that seemed to be promising was the use of 3dfier. The tool allows you to generate smooth road surfaces in different output formats. To implement the practical modeling of roads with 3dfier using the classified point cloud and PICC data, we have followed a few steps. Firstly, the PICC data is initially linear, while 3dfier requires a set of topologically connected polygons as input data. To address this, we use QGIS tools to transform the linear representation into a polygonal result (see Figure 4). The axis and edges provided by the PICC data were transformed into surfaces using QGIS, thereby generating polygons representing the roads. These polygons were then used to perform the lifting based on the semantics of the road polygons. We then adjusted and adapted the 3dfier lifting options and setting parameters. Essentially, 3dfier relies on a binary classification of ground and non-ground (minimum requirements). However, in our case, a detailed classification was performed. We incorporated this detailed classification to accurately extract the "roads" class (use classes 2 and 11), which 3dfier will utilize during the lifting process.
Following the 3dfier requirements, we configured the (.yml) file used in the scope of this work.

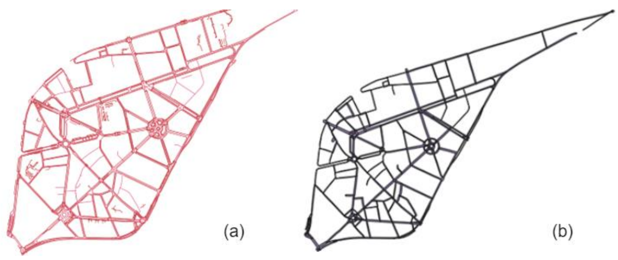

Figure 4. The data preparation for 3dfier road modeling: (a) the shapefile raw data, linear representations (b) the polygonal representation based on QGIS tool.

3.3.3 Automatic vegetation modeling

To automatically generate 3D models of trees from airborne LiDAR point cloud data with a LoD2, a three-step process was followed: classification, segmentation, and modeling. Firstly, the point cloud must be exclusively classified as vegetation, after which individual trees need to be segmented. Finally, these segments of individual trees serve as the data source for constructing 3D tree models. The steps are illustrated in Figure 5.

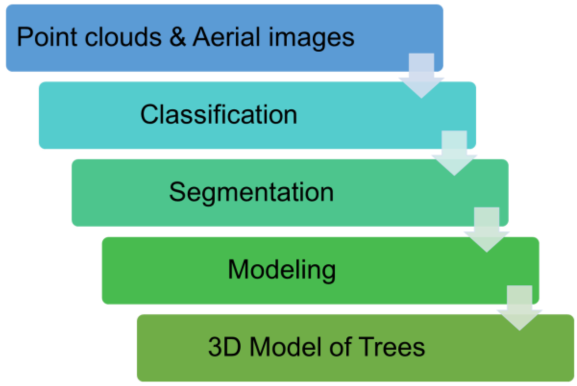

Figure 5. General Workflow for Tree Modeling

A) Classification:

The classification phase has already been detailed in section 3.2. To extract the vegetation point cloud, we utilized the CloudCompare tool. After importing the classified point cloud and displaying the scalar field corresponding to the classification, we proceeded to extract the "vegetation" class. This extraction can be performed in various ways, one of which involves accessing the main menu of CloudCompare and selecting the "Filter by Value" option.

B) Segmentation:

The aim of this step is to assign an identifier to each tree. One can opt to use available automatic codes, such as those presented on (https://github.com/r-lidar/lidR/tree/master), or choose tools like utilizing an algorithm integrated into the CloudCompare tool, as illustrated in this study. To employ CloudCompare, simply access the software's main menu, navigate to "Plugins," and select the "TreeIso" algorithm. Subsequently, we executed three types of segmentation: initial segmentation, intermediate segmentation, and refined segmentation. Adjustments to the parameters were made until achieving a satisfactory result. The selection of parameters depends on the type of data, data quality, etc. Finally, a data cleaning step is crucial, especially in situations where certain trees are not correctly segmented, particularly in dense forest areas. Various automatic or semi-automatic cleaning methods within the CloudCompare tool can be employed.

C) Modeling:

The modeling process was based on the LoD specifications proposed by (Ortega-Córdova 2018). The required parameters are extracted from the segmented vegetation and modeled accordingly. These parameters include the tree top (the 99th percentile of the height), the tree base (ground height), the peripheral point (height range where most points are located), the base of the tree crown (the 5th percentile of the height), and two intermediate divisions for added detail. These divisions are determined using the midpoint between the peripheral point and the top, as well as the base of the tree crown, respectively (de Groot 2020). These parameters are crucial for constructing individual plant objects, as illustrated in Figure 6.

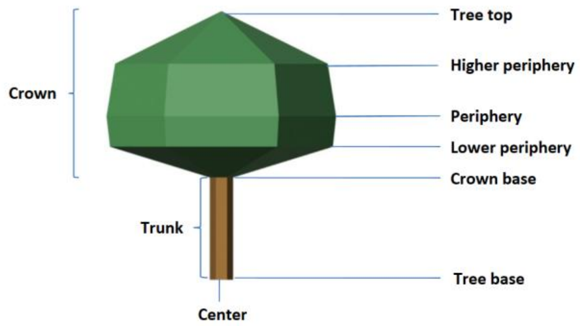

Figure 6. Tree construction parameters (de Groot 2020).

Each LoD employs a different combination of the extracted parameters to construct tree models. LoD0 uses only the peripheral radius and the tree base. LoD1 utilizes the peripheral radius, tree base, and tree top. LoD2 incorporates all the extracted parameters. The 3D tree models are constructed in accordance with CityJSON specifications. It is essential that vertices are arranged in a counterclockwise trigonometric order (CCW) when viewed from the outside, as it is a common rule in 3D modeling. This ensures that the faces have outward oriented normals. This guarantees that the constructed geometry is visible in any rendering software with 3D capabilities and adheres to ISO standards [International Organization for Standardization, 2019].
To initiate the extraction of parameters, the essential input data consists of a point cloud with attributes X, Y, Z, Tree Segment ID (a specific identifier for each tree obtained from the segmentation step), and the attribute "Height Above Ground," which was computed using the ground and tree classes. The calculations were performed using CloudCompare.

4. RESULTS AND DISCUSSION

The resulting 3D model for buildings, roads, and vegetation was compliant with CityJSON v1.1. All models were validated at the schematic and geometric levels. For that, CityJSON has a wide range of free and open-source tools and software that assist and facilitate the use and manipulation of CityJSON data. For instance, Val3dity[9] is an open-source software dedicated to validating the 3D primitives (geometries) of the model. The software reports the geometric and topological errors by specifying the object in concern. For each 3D model, different errors are reported.
The Schema validation was fullfiled based on the official validator for CityJSON files, cjval[10]. Cjio[11] was also used to merge, upgrade, and validate the CityJSON files. The results were visualized using the web viewer ninja[12]. The Table 2 summarizes the validation process results of all city objects modelled in this work.

File	Val3dity	Cjval	Ninja
Buildings.json	93,2%	100% valid	Semantic surfaces
Roads.json	87% valid	100% valid	No semantic surfaces
Vegetation.json	100% valid	100% valid	No semantic surfaces

Table 2. Validation of the different 3D city objects using the open-source validator software.

4.1 3D building model

Geoflow has demonstrated its capabilities in providing good results both in geometric and semantic terms. The building model schema and geometry were both validated. Each building is represented by a specific and unique identifier derived from PICC data. Thus, semantic and attribute information have been accurately assigned to each building. The LoD2.2 is maintained for this work, showing various building elements such as building roofs, walls, etc. We also generated the LoD1.2 and LoD1.3 for future work. This automated modeling method offers significant advantages compared to other existing reconstruction methods in the literature. For instance, we use the same input data to generate the LoD1 building model using the 3dfier tool, which may not be sufficient for certain urban applications. Furthermore, we conducted another approach using FME. We create an FME workbench that produce LoD1 building model. Despite the advantages it offers, this method is semi-automatic and requires human expertise. In addition, it poses challenges in automatically incorporating semantic information. Figure 7 provides an overview result of the three methods. The main errors reported from Val3dity are: Consecutive_Points_Same[13] and Non_Planar_Polygon_Normals_Deviation[14]

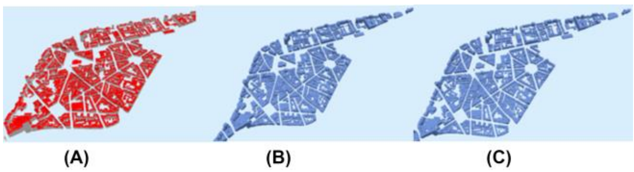

Figure 7. An Example of 3D building model: (A) LoD2 model based on Geoflow, (B) LoD1 based on 3Dfier and (C) LoD1 based on FME of the Outremeuse district.

4.2 Results of 3D Road Modeling

The produced 3D road model from 3dfier (refer to Figure 8) was effectively validated both from geometric and schematic levels. Each road is represented by a unique identifier derived from PICC data; thematic attributes are handled by default by 3dfier. The 3D road model is a LoD1 MultiSurface model. This automated modeling method is relevant while working with 2D polygonal data representation. The errors reported in val3dity are namely: Consecutive_Points_Same and Ring_Self_Intersection[15].
As we explained earlier, we created a FME road workbench as well. However, the result was invalid from a geometric

[9] https://github.com/tudelft3d/val3dity
[10] https://validator.cityjson.org/
[11] https://github.com/cityjson/cjio
[12] https://ninja.cityjson.org/

[13] https://val3dity.readthedocs.io/en/latest/errors/#consecutive-points-same
[14] https://val3dity.readthedocs.io/en/latest/errors/?highlight=Non_Planar_Polygon_Normals_Deviation#non-planar-polygon-normals-deviation
[15] https://val3dity.readthedocs.io/en/latest/errors/?highlight=RING_SELF_INTERSECTION#ring-self-intersection

perspective, and several errors were reported. To solve that, we use triangulate function of Cjio. The obtained model is valid and only Consecutive_Point_Same error was reported for 2% of the 3D primitives.

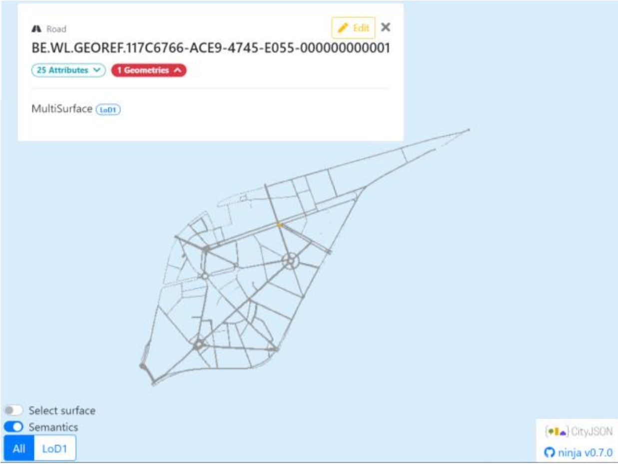

Figure 8. LoD1 road model of the Outremeuse district using 3dfier.

4.3 Results of 3D Vegetation Modeling

The approach employed for tree modeling has yielded satisfactory geometric and semantic results (see Figure 9). The schema and geometry underwent through validation (refer to Table 2). Each tree is represented by a specific identifier, with parameters extracted from the tree and associated semantic information. The level of detail in tree modeling is LoD2, representing a realistic tree form (see Figure 9). This automated modeling approach offers significant advantages compared to other existing reconstruction methods/approaches in the literature. For instance, the use of 3dfier, while advantageous in automatically adding semantic and attributive information, is limited to presenting trees at LoD0, which proves insufficient for certain environmental and ecological applications. The 3D tree model was fully validated, and no geometric errors were reported.
Additionally, employing the tree reconstruction method with FME schemas presents some limitations, including scale issues and the manual addition of semantic information. However, the modeling approach utilized in this study also has its constraints, such as errors in the segmentation step, particularly in densely populated forest areas where precise tree differentiation poses a challenge.

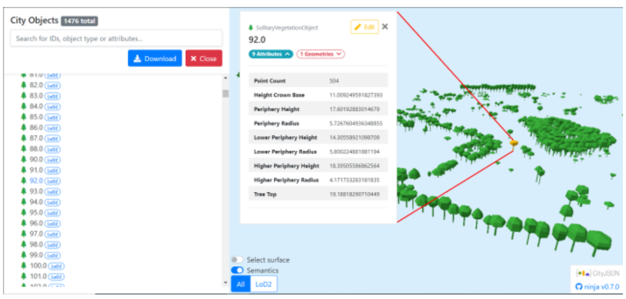

Figure 9. LoD2 tree models of Results of the Outremeuse district.

4.4 Discussion:

The aim of this work was to propose a general and reusable approach to generating 3D city models. The framework ranges from data preparation and pre-processing to 3D modeling. The methodology was implemented in a case study to illustrate the approach and to handle the challenges related to 3D city modeling. Especially since the literature mainly focuses on 3D building modeling, we presented a 3D modeling pipeline for buildings, roads, and vegetation. Table 3 below summarizes the findings according to various criteria. It will help guide the user through the reproducibility and applicability of the process.

3D models	Buildings	Roads	Trees
Type of methods (automatic, semi-automatic, manual)	Automatic	Automatic	Semi-automatic
Input data	PICC data Point Clouds	PICC data Point Clouds	Point Clouds
Minimum required attributes in point clouds	X, Y, Z, Classification	X, Y, Z, Classification	X, Y, Z, Segment ID (for each tree), Height Above Grounds
Point cloud classification (basic or advanced)	Basic	Advanced	Advanced
LoD	LoD2	LoD1	LoD2
License, terms of use of the modeling tool	General Public License	General Public License	Not specified
Supported format (input/output)	Input: point cloud: LAS or. LAZ. 2D polygon: GeoPackage, ESRI Shapefile, or a connection to a PostGIS database Output: CityJSON	Input: point cloud: LAS or. LAZ Output is in the following formats: OBJ, CityGML, CityJSON, CSV (for buildings only, i.e. their ID and height (ground+roof) are output in a tabular format), PostGIS, and STL.	Input: point cloud: LAS or. LAZ. Output: CityJSON
Geometry type	Solid	MultiSurface	MultiSurface
Semantic handling	Yes	No	No
Minimum requirements (configuration file)	Classification into two categories: ground and buildings	Classification into two categories: ground and non-ground	Classification involving two categories: ground

			and trees
Thematic attributes (Native/workaround)	Native support	Native support	-
Time (computational)	Very fast	Very fast	Very fast
Is there any report to guide the user.	Yes: https://github.com/geoflow3d/geoflow-bundle	Yes: https://github.com/tudelft3d/3dfier	Yes: https://github.com/RobbieG91/TreeConstruction

Table 3. Basic information of 3D modeling of the city objects according to various criteria.

Merging 3D buildings, roads, and vegetation could be achieved (refer to Figure 10).

Figure 10. 3D city model of Outremeuse district

5. CONCLUSION

This article presents a processing pipeline designed for the automated modeling of buildings, roads, and vegetation using semantic segmentation outcomes derived from 3D LiDAR point clouds. The methodology employs a good semantic segmentation approach, ensuring precise extraction of target objects. The open-source reconstruction tools for buildings and roads modeling were adapted, and simultaneously, a Python code for tree modeling was optimized. The application of these methods in a case study conducted in Liège, Belgium, yielded satisfactory results, with validated schemas and geometry for the developed models. Furthermore, an evaluation of the adopted reconstruction methods, including a comparative analysis with other techniques from the existing literature, underscores the robustness and efficacy of the proposed approach. As a perspective, we suggest investigating the proposed processing pipeline in other cities that do not yet have an urban model to evaluate its efficiency and limits in different urban contexts. Additionally, we recommend modeling other urban objects with the aim of producing highly detailed urban models rich in urban knowledge.

ACKNOWLEDGMENTS

Funding: This research is part of the SEM 3D project (3D semantic object extraction for urban application) funded by Digital Wallonia-Tremplin AI / Public Service in collaboration with City of Liège.
The authors would like to thank Digital Wallonia, the City of Liège, and the Walloon region for collaborating on the project. We sincerely appreciate the participation of all the experts in the project design and implementation. We highly appreciate all the valuable insights provided for this work.

REFERENCES

Ballouch, Zouhair, Rafika Hajji, Abderrazzaq Kharroubi, Florent Poux, et Roland Billen. 2024. « Investigating Prior-Level Fusion Approaches for Enriched Semantic Segmentation of Urban LiDAR Point ». *Remote Sensing* 16 (janvier). https://doi.org/10.3390/rs16020329.

Ballouch, Zouhair, Rafika Hajji, Florent Poux, Abderrazzaq Kharroubi, et Roland Billen. 2022. « A Prior Level Fusion Approach for the Semantic Segmentation of 3D Point Clouds Using Deep Learning ». *Remote Sensing* 14 (14): 3415. https://doi.org/10.3390/rs14143415.

Batty, Michael. 2018. « Digital Twins ». *Environment and Planning B: Urban Analytics and City Science* 45 (5): 817-20. https://doi.org/10.1177/2399808318796416.

Biljecki, Filip, Jantien Stoter, Hugo Ledoux, Sisi Zlatanova, et Arzu Coltekin. 2015. « Applications of 3D City Models: State of the Art Review ». *ISPRS International Journal of Geo-Information* 4 (décembre): 2842-89. https://doi.org/10.3390/ijgi4042842.

DImitrov, H., et D. Petrova-Antonova. 2021. « 3D City Model as a First Step towards Digital Twin of Sofia City ». In , 43:23-30. https://doi.org/10.5194/isprs-archives-XLIII-B4-2021-23-2021.

Groot, Rob de. 2020. « Automatic Construction of 3D Tree Models in Multiple Levels of Detail from Airborne LiDAR Data ». https://repository.tudelft.nl/islandora/object/uuid%3A3e169fc7-5336-4742-ab9b-18c158637cfe.

Guo, Yulan, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, et Mohammed Bennamoun. 2021. « Deep Learning for 3D Point Clouds: A Survey ». *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43 (12): 4338-64. https://doi.org/10.1109/TPAMI.2020.3005434.

Hu, Qingyong, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, et Andrew Markham. 2020. « RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds ». In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11105-14. Seattle, WA, USA: IEEE. https://doi.org/10.1109/CVPR42600.2020.01112.

Huang, J., J. Stoter, R. Peters, et L. Nan. 2022. « City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds ». *Remote Sensing* 14 (9). https://doi.org/10.3390/rs14092254.

Jeddoub, Imane, Zouhair Ballouch, Rafika Hajji, et Roland Billen. 2024. « Enriched Semantic 3D Point Clouds: An Alternative to 3D City Models for Digital Twin for Cities? » In *Recent Advances in 3D Geoinformation Science*, édité par Thomas H. Kolbe, Andreas Donaubauer, et Christof Beil, 407-23. Lecture Notes in Geoinformation and Cartography.

Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-43699-4_26.

Jeddoub, Imane, Gilles-Antoine Nys, Rafika Hajji, et Roland Billen. 2023. « Digital Twins for Cities: Analyzing the Gap between Concepts and Current Implementations with a Specific Focus on Data Integration ». *International Journal of Applied Earth Observation and Geoinformation* 122 (août): 103440. https://doi.org/10.1016/j.jag.2023.103440.

Ketzler, Bernd, Vasilis Naserentin, Fabio Latino, Christopher Zangelidis, Liane Thuvander, et Anders Logg. 2020. « Digital Twins for Cities: A State of the Art Review ». *Built Environment* 46 (décembre): 547-73. https://doi.org/10.2148/benv.46.4.547.

Khawte, S. S., M. N. Koeva, C. M. Gevaert, S. Oude Elberink, et A. A. Pedro. 2022. « DIGITAL TWIN CREATION FOR SLUMS IN BRAZIL BASED ON UAV DATA ». *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-4/W4-2022 (octobre): 75-81. https://doi.org/10.5194/isprs-archives-XLVIII-4-W4-2022-75-2022.

Ledoux, Hugo, Filip Biljecki, Balázs Dukai, Kavisha Kumar, Ravi Peters, Jantien Stoter, et Tom Commandeur. 2021. « 3dfier: Automatic Reconstruction of 3D City Models ». *Journal of Open Source Software* 6 (57): 2866. https://doi.org/10.21105/joss.02866.

Lehtola, Ville V., Mila Koeva, Sander Oude Elberink, Paulo Raposo, Juho-Pekka Virtanen, Faridaddin Vahdatikhaki, et Simone Borsci. 2022. « Digital Twin of a City: Review of Technology Serving City Needs ». *International Journal of Applied Earth Observation and Geoinformation*, juillet, 102915. https://doi.org/10.1016/j.jag.2022.102915.

Lei, Binyu, Rudi Stouffs, et Filip Biljecki. 2022. « Assessing and benchmarking 3D city models ». *International Journal of Geographical Information Science*, novembre. https://doi.org/10.1080/13658816.2022.2140808.

Masoumi, Homa, Sara Shirowzhan, Paria Eskandarpour, et Christopher James Pettit. 2023. « City Digital Twins: their maturity level and differentiation from 3D city models ». *Big Earth Data* 0 (0): 1-46. https://doi.org/10.1080/20964471.2022.2160156.

Nys, G.-A., R. Billen, et F. Poux. 2020. « AUTOMATIC 3D BUILDINGS COMPACT RECONSTRUCTION from LIDAR POINT CLOUDS ». In , 43:473-78. https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-473-2020.

Ortega-Córdova, Lessie. 2018. « Urban Vegetation Modeling 3D Levels of Detail ». https://repository.tudelft.nl/islandora/object/uuid%3A8b8967a8-0a0f-498f-9d37-71c6c3e532af.

Peters, Ravi, Balázs Dukai, Stelios Vitalis, Jordi van Liempt, et Jantien Stoter. 2022. « Automated 3D Reconstruction of LoD2 and LoD1 Models for All 10 Million Buildings of the Netherlands ». *Photogrammetric Engineering and Remote Sensing* 88 (3): 165-70. https://doi.org/10.14358/PERS.21-00032R2.

Stoter, J. E., GAK Arroyo Ohori, B. Dukai, A. Labetski, K. Kavisha, S. Vitalis, et H. Ledoux. 2020. « State of the art in 3D city modelling: Six challenges facing 3D data as a platform ». *GIM International: the worldwide magazine for geomatics* 34.

Stoter, J.E., G.A.K. Arroyo Ohori, et F. Noardo. 2021. « Digital Twins: A Comprehensive Solution or Hopeful Vision? » *GIM International: the worldwide magazine for geomatics* 2021 (October).

Yarroudh, A., G.-A. Nys, et R. Hajji. 2023. « 3D MODELING OF ROAD INFRASTRUCTURES ACCORDING TO CITYGML 3.0 AND ITS CITYJSON ENCODING ». *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-1-W2-2023 (décembre): 63-70. https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-63-2023.

APPENDIX

FME workbench of road modeling