Automatic Transformation of Semantic 2D Lane Models into 3D CityGML Representations

Zihan Deng[1], Ihab Hijazi[2, 1], Christof Beil[1], Thomas H. Kolbe[1]

[1] Chair of Geoinformatics, Technical University of Munich, Arcisstr. 21, Munich, Germany - (zihan.deng, ihab.hijazi, christof.beil, thomas.kolbe)@tum.de

[2] Urban Planning Engineering Department, An-Najah National University, Nablus, Palestine - eehab@najah.edu

Keywords: 2D to 3D transformation, Semantic 3D Streetspace, Semantic information, CityGML.

Abstract

Urban digital twins are becoming essential for transportation applications, demanding precise geometric, semantic, and topological data. However, existing transportation infrastructure information is typically available in 2D formats, while many applications require accurate 3D representations. Existing 3D representations, such as point cloud data, often lack integrated semantic information. This paper addresses this gap by presenting a novel method for the automatic transformation of semantic 2D lane models into 3D CityGML representations. The transformation process comprises three main phases: (1) Point cloud data processing: Noise and irrelevant structures are removed, retaining essential 3D lane features, and elevation information is derived by converting the point cloud data into digital elevation models (DEMs); (2) Segmentation and smoothing: Extracted DEMs undergo segmentation, noise removal, and refinement to ensure geometric continuity; and (3) Transformation and postprocessing: The semantic 2D lane models are integrated with the processed DEMs through elevation interpolation, followed by refinement and transformation into 3D CityGML representations. Compared to existing methods, the proposed method delivers more realistic and comprehensive 3D lane models while maintaining efficiency. A case study in Munich, Germany, demonstrates the algorithm's effectiveness in addressing challenges in complex scenarios including tunnels and bridges. The paper concludes by discussing encountered challenges and proposing future research directions to advance the integration of 2D and 3D transportation infrastructure information.

1. Introduction

Semantic 3D streetspace models have become essential components in the transportation sector. They ensure consistency and completeness within semantic 3D city models. Geometrically, these models enhance the continuity and coherence of urban representations, facilitating more efficient infrastructure planning, management, and maintenance processes (Williams et al., 2013). From a semantic perspective, they extend beyond traditional visualization applications, providing actionable insights that support advanced technologies such as autonomous driving and traffic simulation (Ma et al., 2018). By integrating detailed geometric, topological, semantic, and temporal data, semantic 3D streetspace models are reshaping how transportation infrastructure is analyzed and managed (Beil and Kolbe, 2024). Moreover, as urban areas expand and grow increasingly complex, there is a notable shift from focusing solely on horizontal space utilization to the incorporation of vertical spaces. This evolution highlights the need for detailed modeling of streetspaces in complex scenarios, such as tunnels and bridges, which are now critical topics in semantic 3D streetspace modeling. These developments reflect the growing demand for comprehensive urban representations capable of capturing the intricacies of modern urban infrastructure. Consequently, semantic 3D streetspace models are advancing to meet these demands, playing a pivotal role in addressing the complexities of urban environments and supporting the sustainable development of cities.

Geometrically, 3D streetspace is frequently reconstructed within virtual environments to ensure detailed and reliable urban representations. There are two primary methods for representing 3D information: direct and indirect (Chen et al., 2022). The direct method utilizes technologies such as laser scanners to capture intricate 3D geometric data, providing an accurate representation of real-world environments. Among these, LiDAR stands out for its ability to measure distances using laser pulses, generating highly precise spatial data (Gil et al., 2013). Moreover, advanced mapping systems like the Trimble MX9 (Trimble, 2024) and NavVis VLX (NavVis, 2024) further enhance precision by delivering detailed point cloud data. These advancements in technology now play a crucial role in various applications, such as enhancing pedestrian accessibility mapping, which contributes to safer and more accessible urban mobility solutions (González-Collazo et al., 2024). In contrast, the indirect method represents 3D information through models such as digital elevation models (DEMs) (Hirt, 2015). DEMs are often used as a generic term encompassing both digital terrain models (DTMs) and digital surface models (DSMs). DTMs, typically derived from LiDAR, satellite, or aerial data, represent the bare earth terrain by excluding above-ground features like buildings (Li et al., 2005), while DSMs include all natural and man-made objects above the ground. Both DTMs and DSMs are often stored as 2.5D raster data, with elevation information encoded in each grid cell, offering an efficient yet distinct method of representing 3D urban environments.

While the geometric representation of 3D streetspace is a crucial aspect, it is only one part of a broader entity. Semantic information is essential for enriching contextual data, enabling a comprehensive understanding of the environment. With advancements in deep learning, semantic segmentation of point cloud data has been applied to streetspace modeling. However, the results are limited to predefined categories. Additional details, such as material composition or speed limits within the streetspace, cannot be identified using these methods. Streetspace semantic information is often available from government agencies or through paid services, such as Google Maps (Google Earth Engine Team, 2024). Additionally, many

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W6-2025-57-2025

governments offer online mapping platforms with detailed semantic information about 3D streetspace. In Germany, BayernAtlas (BayernAtlas Team, 2020) offers comprehensive information about Bavaria, including street networks and classifications. Meanwhile, the City of Munich is working to standardize and collect uniform data to create more detailed representations of semantic lane models in streetspace.

Despite the increasing availability of detailed semantic information, most government-provided data remains confined to 2D geometries, such as 2D polygons and 2D networks. However, many applications of semantic streetspace models require 3D representations for accurate analysis and simulation. Thus, it is imperative to transform rich semantic 2D streetspace models into semantic 3D streetspace models. A comprehensive semantic framework further improves the representation of diverse transportation types. To achieve this, transformed models must comply with established standards, such as the OGC international standard CityGML, ensuring consistency and interoperability in semantic 3D city modeling. The latest version, CityGML 3.0, introduces an enhanced transportation module to reduce redundant geometric representations and ensure the continuity and completeness of semantic 3D streetspace models (Beil, 2025).

This paper proposes a novel method to transform semantic 2D lane models developed by City of Munich into realistic semantic 3D lane models. The method effectively manages complex scales while preserving all semantic information, ensuring compliance with the standardized CityGML data model. The method provides a robust foundation for essential downstream applications within urban digital twin systems, enhancing functionality and integration across diverse urban planning and transportation use cases (Beil and Kolbe, 2024).

The remainder of this paper is organized as follows: Section 2 presents a review of the research background. Section 3 details the proposed method. Section 4 describes the experiments conducted and their implementation. Finally, Section 5 presents the study's conclusions and outlines potential directions for future research.

2. Related work

Current research in semantic 3D streetspace modeling can be broadly categorized into two approaches. The first approach relies on existing 3D geometric representations to derive corresponding semantic information, which this paper defines as the Bottom-Up approach. Conversely, the second approach is based on GIS data and employs rule-based modeling algorithms to generate semantic 3D streetspace models, which this paper defines as the Top-Down approach.

2.1 Bottom-Up approach

Numerous methods have been developed to generate semantic 3D models by segmenting point cloud data into distinct components to derive semantic information (Borisov et al., 2022). In the transportation sector, Manandhar and Shibasaki developed an algorithm for extracting road surfaces based on features such as point density, elevation, and slope from point cloud data (Manandhar and Shibasaki, 2002). This approach was further extended to achieve high-accuracy extraction of road markings by leveraging their retro-reflective properties (Kumar et al., 2014). Additionally, methods for extracting curbs have been proposed to support transportation applications (Zai et al., 2017). The integration of imagery data has also proven highly effective. Gao et al. combined LiDAR point clouds with high-resolution remote sensing imagery to extract 3D roads in Hong Kong (Gao et al., 2021). Moreover, machine learning and deep learning techniques are increasingly being applied. In built environments, Robert et al. introduced a superpoint-based transformer architecture that achieved high accuracy in detecting ground, vehicles, vegetation, and other features (Robert et al., 2023). Additionally, Pan et al. segmented road pavements into finer components, such as road shoulders, and represented them hierarchically using graph structures (Pan et al., 2024). However, the Bottom-Up approach faces several challenges in semantic 3D streetspace modeling. While point cloud data can effectively capture real-world conditions, existing methods still struggle with accurately segmenting fine-grained and small features. Additionally, handling complex scenarios, such as bridges and tunnels, remains difficult, as these structures may contain overlapping 3D points or occluded areas that hinder precise segmentation. Furthermore, semantic segmentation is often restricted to predefined categories, making it challenging to derive the detailed and application-specific information required for downstream applications in urban digital twins.

2.2 Top-Down approach

GIS data is highly valuable for semantic 3D streetspace modeling, with open-source DEM datasets such as the Shuttle Radar Topography Mission (SRTM) providing global coverage (Tachikawa et al., 2011). Additionally, many regions, such as Bavaria in Germany, have made official GIS data publicly available (BayernAtlas Team, 2020). Building on such datasets, Wang et al. developed a method to extract road information using remote sensing images integrated with terrain models to render 3D road networks (Wang et al., 2018). However, this method struggled with modeling accuracy. To address this, Zhang et al. proposed a scalable, template-based approach for 3D road modeling that enhances precision, making it suitable for larger areas (Zhang et al., 2019). For more detailed data and the ability to handle complex terrains, Wang et al. introduced a method for automatically generating large-scale 3D road networks by integrating GIS data with road semantic information (Wang et al., 2021). However, their approach did not adequately model streetspaces in tunnels or bridges. To overcome this, Chen et al. introduced a rule-based method for generating 3D road networks that accurately represent these complex areas using open-source data (Chen et al., 2022). The Top-Down approach benefits from greater data accessibility by leveraging GIS data, making it suitable for large-scale modeling. However, such models often do not accurately capture real-world road characteristics, such as width and surface texture. Additionally, they do not preserve or extract valuable semantic information that is sufficiently detailed or applicable for downstream applications.

2.3 Summary and identified research gaps

Numerous studies have explored semantic 3D streetspace modeling, yet several challenges remain. First, many studies struggle to define and derive a sufficient number of useful semantic information for urban planning and applications such as traffic simulation, including attributes like speed limits. Additionally, segmentation accuracy remains a significant issue in real-world modeling. Moreover, existing methods have

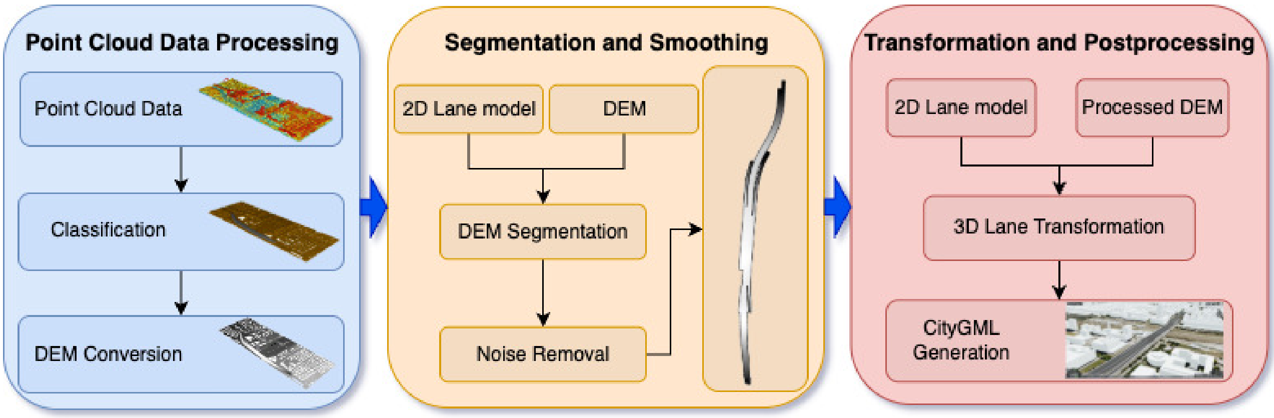


Figure 1. Workflow for 3D lane model transformation.

not managed semantic 3D streetspace models using a universally accepted standard for different use cases. While OpenDRIVE (Schwab and Kolbe, 2022) and IfcRoad (buildingSMART, 2024) provide standards for streetspace models management, they are not fully compatible with GIS systems, limiting their integration and usability for city mapping agencies and other downstream applications. The City of Munich has developed structured semantic 2D lane models by integrating various types of semantic information into 2D polygons, such as speed limits and lane types. At the same time, these 2D polygons precisely represent the geometric shape of the lane model. This approach effectively addresses challenges related to semantic classification and segmentation accuracy. However, since the model is limited to 2D, mapping it to real-world 3D conditions remains a significant challenge. We introduce a novel method to overcome these limitations by transforming developed semantic 2D lane models into 3D representations. The method can effectively handle noise and outliers while preserving detailed semantic information. This method ensures a realistic representation by accurately capturing the shape, boundaries, and spatial extent of semantic 3D lane model. Furthermore, by integrating the CityGML 3.0 data model, it enhances efficiency, accuracy, and completeness, providing a robust and scalable solution for urban digital twin applications.

3. Methodology

In Munich, semantic 2D lane models have been developed and visualized to provide a comprehensive and precise representation of streetspace. These models are based on extensive survey data and existing geospatial information. The models consist of multiple 2D polygons, each accurately depicting its corresponding position of the lane. Additionally, each polygon contains rich semantic information, including lane type, material, elevation level, and other relevant attributes. The semantic 2D lane models already provide information at multiple elevation levels. For instance, elevation level 1 represents elevated structures such as bridges, level 0 corresponds to ground level lanes where height information aligns with the terrain, and level -1 denotes underground structures. Figure 2 shows an example of a section of the semantic 2D lane models.

The objective of this research is to transform semantic 2D lane models into a 3D CityGML representation using existing 3D data sources (e.g., point cloud data and digital elevation models) and semantic information stored in 2D lane models (e.g., elevation levels and lane types). As illustrated in Figure 1, the method consists of the following key steps:

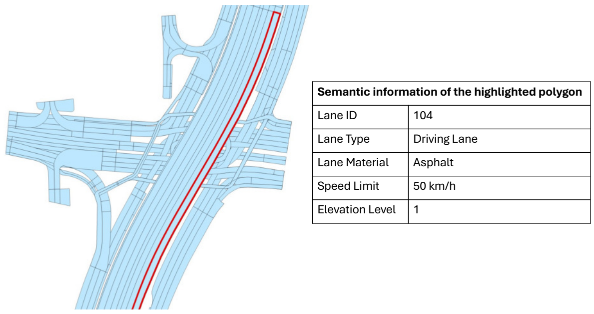


Figure 2. A section of the semantic 2D lane model with an illustration of the semantic information stored within the highlighted polygon.

1. **Point Cloud Data Processing**: The point cloud data is classified using a deep learning point cloud classification model. The relevant points are extracted from the point cloud, and the data are then converted into a digital elevation model (DEM).

2. **Segmentation and Smoothing**: The boundary is determined using the semantic 2D lane models, enabling the extraction of the relevant area from the DEM. Once extracted, a smoothing algorithm is applied to the DEM.

3. **Transformation and Postprocessing**: The semantic 2D lane models are fused with the processed DEM to create a semantic 3D lane model. The final output is then transformed into the CityGML data model.

3.1 Point cloud data processing

In the first step, we extract the 3D ground from the point cloud data. During this process, the point cloud data is classified into different classes using a pre-trained model that assigns each point cluster to predefined classes. To streamline the pipeline within a unified environment, we used a pre-trained model based on the PointCNN architecture available in the GIS software. These classes include ground, buildings, and others. In our study, the ground class represents the lane. All other classes are considered noise and are removed. The remaining points representing the ground are then converted into a 2.5D raster format with embedded elevation information. Since the

3D points include elevation values, the 3D to 2.5D conversion extracts and embeds this information into the 2.5D representation, with each grid cell storing the elevation data for its corresponding location. The process is illustrated in Figure 3.

However, for semantic 3D lane modeling, basic category-level classification is insufficient to meet the requirements of high-quality 3D modeling. Despite advancements in deep learning model accuracy, challenges persist, including noise, misclassifications, and empty spots in the point cloud data. Additionally, the 2D lane model boundaries often do not align perfectly with the point cloud data, resulting in boundary noise. Thus, after removing irrelevant points, the resulting raster may contain voids and errors due to the mislabeled points. Using such DEMs directly for 3D modeling poses significant issues, as the process is highly sensitive to elevation model quality. Given that many downstream applications of semantic 3D lane models demand fine-grained details, we further process the raster to enhance its accuracy, ensuring it meets the precision standards necessary for semantic 3D modeling.

3.2 Segmentation and smoothing

This step aims to address the errors present in the DEM generated from the point cloud data. The first phase is DEM segmentation. Since point cloud data encompass extensive urban areas, as shown in Figure 3c, the resulting 2.5D elevation model may include not only the lane but also surrounding areas that are irrelevant for the transformation. To resolve this, we leverage the semantic 2D lane models. We delineate the boundary of the target area automatically using polygon data from the 2D lane model provided by the City of Munich for conversion and extract the relevant portion of the DEM within the semantic 2D lane model boundary. This ensures that only the intended street-space is included in the transformation process.

Next, the extracted DEM undergoes a smoothing process. This step addresses three key challenges: 1. It fills the voids in the DEM caused by the removal of structures from the point cloud data. 2. It removes errors and noise, particularly along the boundaries where the 2D lane model may not fully align with the point cloud data, resulting in boundary inaccuracies. 3. It ensures compliance with road design standards, such as the maximum allowable gradient for roads. For example, in Germany, the maximum gradient for main roads in built-up areas is typically 6% (Baier, 2024)(Hartkopf, 2024).

The algorithm begins by checking the slope of the raster. For each grid cell in the raster, the slope is calculated based on the elevation values of its neighboring grid cells, generating a slope raster that represents the slope for each location. If the slope raster cell exceeds the predefined threshold (e.g., 6%), the elevation raster cell is treated as erroneous in terms of elevation accuracy and is marked as void. After removing these erroneous cells, the resulting raster adheres to the standard but contains many voids due to slope removal and point cloud classification. To address this issue, we use an interpolation function, Spline (Cao et al., 2009), to estimate missing values based on the spatial distribution of known values. Next, map algebra (Jeremy Mennis and Tomlin, 2005) further smooths the surface. By computing the mean values of eight neighboring cells with equal weighting, map algebra helps generate a more accurate surface. After iterating this process twice, the raster shows significant improvement compared to the initial version, as shown in Figure 4a and 4b. However, even with these improvements, the result may still fall short of the required standards. The interpolation and map algebra processes cannot perfectly predict or fill boundary grid cells, as they often lack sufficient neighbors for reference. Therefore, we perform another slope check to ensure that all grid cells in the raster comply with road construction standards, as shown in Figure 4c.

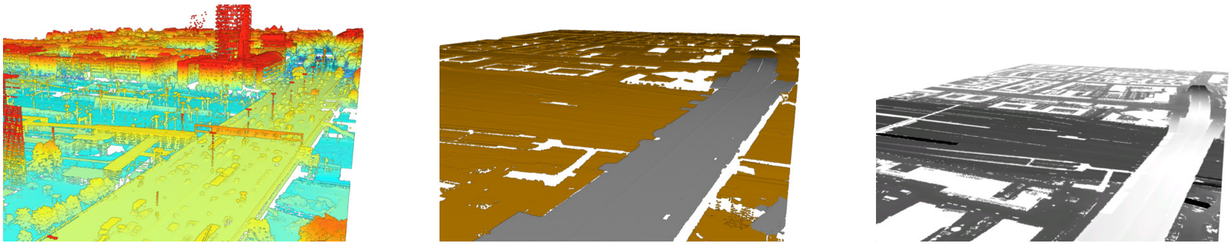
To address the remaining voids in the DEM, we employ Thiessen polygons to further fill missing cell values based on their nearest points. Each polygon is assigned the attribute of its corresponding point, ensuring that all areas within the polygon share the same attribute. As illustrated in Figure 5, the process begins by converting the processed raster into points, where each point holds the elevation value of its corresponding grid cell. Next, Thiessen polygons are generated, constrained by the 2D lane models boundary to ensure polygons are created only within the defined region. These polygons are then converted back into points, which are positioned at the vertices of the polygons and assigned the elevation information of their respective polygons. Using these points, a new elevation surface is generated as a Triangulated Irregular Network (TIN).

Algorithm 1 Semantic 2D Lane Model to Semantic 3D Multi-Polygon Transformation

Input: P_{mobile}, P_{aerial}, DTM, LM_{2D}
Parameters: $n \leftarrow 2$, $\theta_{max} \leftarrow 6\%$
Output: LM_{3D}
for each lane $l \in LM_{2D}$ **do**
  if $l.level = 1$ **then** ▷ Above-ground: use aerial data
    $DEM_l \leftarrow process(P_{aerial}, DTM)$
  else if $l.level = 0$ **then** ▷ Ground-level: use DTM
    $DEM_l \leftarrow DTM$
  else ▷ Level -1: tunnels, use mobile data
    $DEM_l \leftarrow process(P_{mobile})$
  end if
  $RG_l \leftarrow segment(DEM_l)$
  for $i = 1$ **to** n **do**
    $RG_l \leftarrow removeCells(RG_l, \theta_{max})$
    $RG_l \leftarrow interpolate(RG_l)$
    $RG_l \leftarrow smooth(RG_l)$
  end for
  $RG_l \leftarrow removeCells(RG_l, \theta_{max})$
  $ThieP \leftarrow convertToThiessenPolygons(RG_l, LM_{2D}.boundary)$
  $ThiePoints \leftarrow extractVertices(ThieP)$
  $TIN_l \leftarrow createTIN(ThiePoints)$
  $TIN_l \leftarrow postProcess(TIN_l, LM_{2D}.boundary)$
  $LM_{3D}^{(l)} \leftarrow adapt3D(TIN_l, l)$
end for
$LM_{3D} \leftarrow alignTopologically(\{LM_{3D}^{(l)}\})$
return LM_{3D}

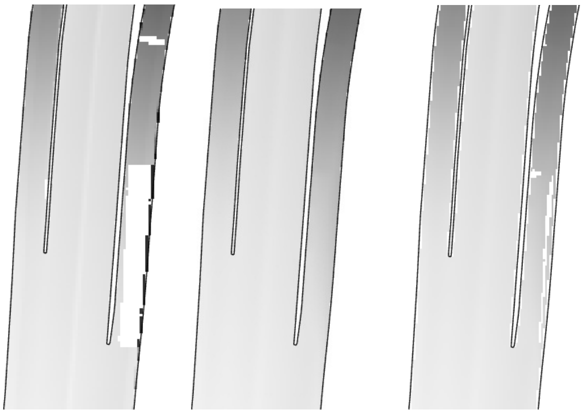
3.3 Transformation and Postprocessing

Postprocessing of the generated TIN is necessary to better align elevation information with the semantic 2D lane models and to mitigate distortions on the lane surface. Since the TIN is generated from a large number of points, the elevation values on the lane surface may be unevenly distributed, leading to visible distortions that affect the model's appearance and accuracy. To address this, it is more effective to use only the elevation data from the polygon boundaries, ensuring a smoother and more visually consistent lane surface. In this process, the semantic 2D lane models are converted into points along the polygons' boundaries, with each point assigned an elevation value based on its corresponding position in the generated TIN. A new TIN is then generated with those points, containing only the elevation surface information from the polygon boundaries, resulting in a visually refined lane model.



(a) Original point cloud data (b) Classified and cleaned point cloud data (c) Converted DEM

Figure 3. Illustration of Point Cloud Data Processing: Image (a) presents the original point cloud data, color-coded by elevation, including structures such as bridges and buildings. Image (b) displays the classified and cleaned point cloud color-coded by classes, where non-ground structures have been removed. Image (c) illustrates the converted Digital Elevation Model, representing the ground elevation, where darker grid cells indicate lower elevations and lighter grid cells represent higher elevations.

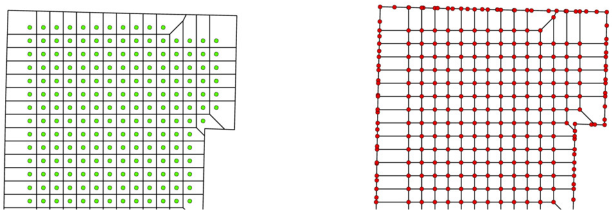


(a) Lane DEM (b) Processed DEM (c) Slope checked

Figure 4. Visualization of the DEM smoothing process, where darker colors represent lower height values and vice versa. Image (a) depicts the segmented lane DEM with voids and noise. After applying the smoothing procedure, image (b) shows a refined DEM with voids and noise removed. However, after the slope check, some areas still did not meet standards and were removed, as seen in image (c).

Finally, 3D multi-polygons are created by interpolating elevation information from the newly generated TIN surface to semantic 2D polygon features. To handle complex scenarios such as bridges and tunnels, the semantic information particularly the elevation levels provided in the semantic 2D lane models helps address this challenge. Different elevation levels correspond to different 3D representations during the transformation process. For example, for level 1 polygons, elevation information can be derived from aerial point cloud data; for level 0 polygons, digital terrain models can be used directly; and for level -1 polygons, elevation values are matched with mobile mapping point cloud data from tunnels. Once the corresponding semantic 2D lane models and their respective 3D representations are established, each level's elevation information can be automatically interpolated into the corresponding semantic 2D lane models to facilitate the transformation.

After the transformation, the 3D multi-polygons for different levels are connected to ensure a continuous representation. Theoretically, 3D intersection lines between different elevation levels should maintain consistent elevation values to ensure geometric continuity across the entire model. To enforce this, 3D intersection lines are identified based on the 2D lane model. They are decomposed into points, each associated with its respective elevation. To enforce consistency, the elevation values of these points are adjusted to a common reference elevation that represents the shared boundary between adjacent levels.



(a) Generation of Thiessen Polygons (b) Generation of Vertices Points

Figure 5. Visualization of Thiessen Polygons. Image (a) illustrates the generation of Thiessen polygons from points derived from the DEM. The green points represent the DEM-derived points, which contain elevation information. The Thiessen polygons are constructed based on the spatial distribution of these points, with each polygon sharing the elevation information of its corresponding point. In image (b), the red points represent the vertices of each polygon, which inherit the elevation information from their respective polygons.

Each step of the method can be integrated into a single function, enabling the entire process to be executed as an automated transformation. The required inputs include the different elevation levels of 3D representations and the corresponding semantic 2D lane models. All these steps can be combined into a unified workflow within a Python script, allowing for seamless integration of previously implemented methods. Alternatively, the process can be visualized and executed using model-building tools within a GIS software environment. The complete workflow is summarized in Algorithm 1.

The result of the transformation process is in the form of 3D multi-polygons. These 3D multi-polygons support conversion to the CityGML representation, which aids in managing both geometric and semantic information. The 3D representation contains various semantic attributes inherited from the semantic 2D lane models, such as lane ID, usage, and other relevant information. CityGML, specifically the CityGML transportation module, has already defined the usage and classification for different types of lane models (Beil and Kolbe, 2020). By associating the semantic information embedded within the lane model polygons, the 3D multi-polygons can be further refined into various feature classes and transformed into the CityGML data model.

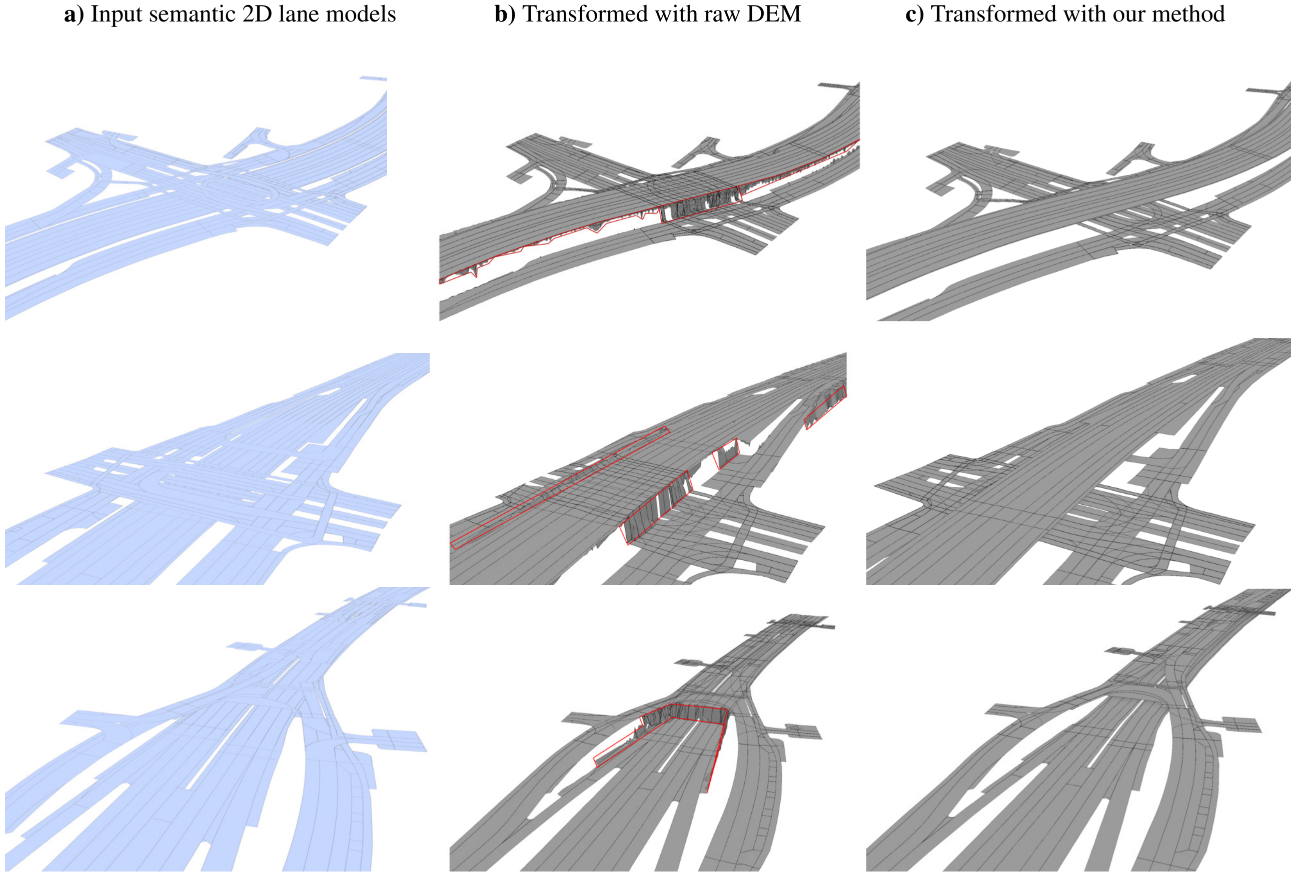


Figure 6. Transformation results for the Donnersberger Bridge and Trappentreu Tunnel case studies. Column a) shows the semantic 2D lane models, column b) shows the results using raw DEM, and column c) shows the results of our method. Distortions in the raw DEM are highlighted in red for comparison.

4. Experiment

4.1 Implementation details

The test was conducted on a computer equipped with an RTX 3070 GPU (6GB VRAM), an Intel i7 CPU with 16 cores, a base frequency of 3.2GHz, and 16GB RAM. Our algorithm was developed in Python, utilizing the ArcPy library, a comprehensive Python package for geographic data analysis within the ArcGIS environment. Besides, by leveraging the ModelBuilder tool in ArcGIS Pro, we visualized the process and interconnected its components to create a complete pipeline. For the classification of point cloud data, we used Esri's pretrained deep learning models based on the PointCNN architecture, implemented within the ArcGIS API for Python (Esri, 2024). The final transformation to CityGML LoD2 transportation module features was carried out using FME software. For web-based visualization, we employed the 3DCityDB web client. The model was transformed into Cesium 3DTiles using FME and then imported into the 3DCityDB web client, enabling efficient visualization and interaction within a 3D web environment. We focused on two commercial tools, ArcGIS Pro and FME, due to their tightly integrated workflows. ArcGIS Pro combines key functions like point cloud classification and DEM processing in one environment, streamlining development. FME's robust support for CityGML LoD2 and 3DTiles generation greatly accelerated exports and reduced development time.

The case study was conducted for the Donnersberger Bridge and Trappentreu Tunnel area in Munich, Germany. This is a highly complex traffic hub, featuring bridges, tunnels, viaducts, and multiple intersections, connecting the northern and southern parts of Munich. The input data includes a semantic 2D lane models that accurately represents lane shapes while providing essential semantic information. We also utilized aerial point cloud data in LAS format with a density of 6 points per square meter, covering the area from Donnersberger Bridge to Trappentreu Tunnel. It includes highly detailed mobile mapping point cloud data with a density of 2000 points per square meter specifically collected for the Trappentreu Tunnel. Additionally, a Digital Terrain Model data with a 1-meter resolution was employed for the case study area. For the transformation of Level 0, which represents the ground level, the DTM was used directly as it already provides pure ground elevation.

The results of the automatic transformation are presented in Figure 6. The left column of figures shows the semantic 2D lane models. To assess performance, we compare two transformation approaches: direct transformation from the raw DEM (generated using point cloud data) and our proposed transformation method. As shown in the middle column of figures, direct transformation from the raw DEM results in significant distortions due to misalignment between the semantic 2D lane models and the 3D point cloud data. In addition, it is not equipped to handle multi elevation level modeling for complex area. The significant distortions are highlighted in red. In contrast, our method, shown in the right column of figures, produces a smoother surface with no visible distortions and seamless connections between different levels, demonstrating the effectiveness of our processing step in improving transformation quality.

(a) Bridge Entrance (b) Main Bridge (c) Bridge Exit

(d) Viaduct Section (e) Tunnel Entrance (f) Tunnel Exit

Figure 7. Visualization of the semantic 3D lane model in the 3DCityDB web client. Images (a) to (f) show different parts of the Donnersberger Bridge and Trappentreu Tunnel.

Table 1. Error rate comparison based on slope accuracy

Metric	Raw DEM Transformation	Our Transformation
Total slope points	104,437	91,979
Error slope points	10,098	213
Error rate	9.67%	0.23%

To evaluate the accuracy of the transformation, we defined an algorithm based on the slope of the 3D multi-polygon lane models. As mentioned earlier, in Germany, the maximum allowable gradient for main roads is 6%. Using this standard, we assessed how much of the transformed lane surface exceeds the threshold. First, the 3D multi-polygon models are converted into a 2.5D raster format, where each grid cell represents the corresponding elevation. Next, the slope is calculated from this elevation raster, resulting in a new raster where each cell contains the slope value derived from its surrounding neighborhood. This slope raster is then converted into a set of points, with each point inheriting the slope value of its corresponding grid cell. We compared the slope information of the results obtained from two methods. As shown in Table 1, the raw DEM transformation resulted in an error rate of 9.67%. In contrast, the proposed method significantly reduced the error to just 0.23%, demonstrating a substantial improvement in accuracy. Finally, the transformation result was converted into CityGML and subsequently transformed into 3DTiles format for web visualization using the 3DCityDB Web Client. Interaction with the components confirms that semantic information was successfully preserved, as shown in Figure 7.

5. Conclusions

In conclusion, our research presents a novel algorithm and workflow for semantic 3D lane model generation by transforming semantic 2D lane models using 3D representations (e.g., point cloud data and digital terrain models) to create a semantic 3D lane model. Throughout the process, point cloud data is classified and converted into a 2.5D elevation model. After segmentation and smoothing, it is fused with the semantic 2D lane models to generate a semantic 3D CityGML representation. The key contributions of our research are as follows:

Firstly, compared to earlier methods, our approach addresses the challenges of modeling complex scenarios. Secondly, our approach fully preserves both the geometric details and the semantic information of the lane model. This is particularly valuable for downstream applications. Lastly, the automated algorithm holds significant potential for large-scale 3D city modeling by improving both the efficiency and speed of the modeling process. Despite these contributions, our approach has some limitations. Firstly, when lane models are too narrow, there are insufficient neighboring cells for smoothing, leading to distortions in the final result. Besides, it should be noted that most of the steps in our workflow depend on commercial software tools, which may limit full reproducibility for users without access to these products. Furthermore, matching point cloud data to lane levels still relies on manual visualization and identification, leading to increased effort in complex multi-level scenarios. Finally, due to limited data and evaluation methods, the assessment is not yet sufficiently comprehensive.

In the future, we plan to test the pipeline on a larger-scale area and develop an algorithm for automatically assigning point cloud data to lane levels. We also aim to explore and integrate more open-source tools to build a more scalable and reproducible workflow. Additionally, with results from broader case studies, we will conduct a more comprehensive performance assessment. Our method works well for 'lane' and 'area' levels but faces challenges with 'way' models. Future work will improve neighbor-weighting to better handle 'way' granularity.

Acknowledgments

We sincerely thank the City of Munich for their collaboration in the Connected Urban Twins (CUT) project. Their support in providing data and facilitating the case study has been instrumental in the successful implementation of this research. We also extend our appreciation to the student members of Group A from the practical course 'Fusion Lab' in the master's program 'Information Technologies for the Built Environment' at TUM for their valuable insights to this research.

References

Baier, 2024. Raa: Guidelines for road construction in germany. https://www.fgsv-verlag.de/pub/media/pdf/200.i.pdf. (Last Accessed: 07 September 2024).

BayernAtlas Team, 2020. Bayernatlas. https://atlas.bayern.de/. (Last accessed: 1 December 2024).

Beil, C., 2025. Detailed streetspace modelling in the context of semantic 3D city and landscape models. PhD thesis, Technische Universität München.

Beil, C., Kolbe, T. H., 2020. Combined modeling of multiple transportation infrastructure within 3D city models and it's implementation in CityGML 3.0. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, VI-4/W1-2020, 29–36.

Beil, C., Kolbe, T. H., 2024. Applications for Semantic 3D Streetspace Models and Their Requirements—A Review and Look at the Road Ahead. *ISPRS International Journal of Geo-Information*, 13(10).

Borisov, M., Radulović, V., Ilić, Z., Petrović, V. M., Rakićević, N., 2022. An Automated Process of Creating 3D City Model for Monitoring Urban Infrastructures. *Journal of Geographical Research*, 5(2), 1–10.

buildingSMART, 2024. IfcFacility: Infrastructure Domain Extension. https://ifc43-docs.standards.buildingsmart.org. (Last accessed: 1 March 2025).

Cao, W., Hu, J., Yu, X., 2009. A study on temperature interpolation methods based on gis. *2009 17th International Conference on Geoinformatics*, 1–5.

Chen, Y., Yang, X., Yang, L., Feng, J., 2022. An Automatic Approach to Extracting Large-Scale Three-Dimensional Road Networks Using Open-Source Data. *Remote Sensing*, 14(22), 5746.

Esri, 2024. Arcgis pretrained deep learning models. Access: https://www.esri.com/en-us/arcgis/deep-learning-models. (Last accessed: 1 February 2025).

Gao, L., Shi, W., Zhu, J., Shao, P., Sun, S., Li, Y., Wang, F., Gao, F., 2021. Novel Framework for 3D Road Extraction Based on Airborne LiDAR and High-Resolution Remote Sensing Imagery. *Remote Sensing*, 13(23).

Gil, A. L., Núñez-Casillas, L., Isenburg, M., Benito, A. A., Bello, J. J. R., Arbelo, M., 2013. A comparison between LiDAR and photogrammetry digital terrain models in a forest area on Tenerife Island. *Canadian Journal of Remote Sensing*, 39(5), 396–409.

González-Collazo, S. M., Balado, J., Túñez Alcalde, R. M., Garrido, I., Lorenzo, H., Díaz-Vilariño, L., 2024. Enhancing Urban Pathfinding for Pedestrians through Fusion of MLS and HMLS Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-4/W9-2024, 197–204.

Google Earth Engine Team, 2024. Google earth engine: A planetary-scale platform for environmental data analysis. https://earthengine.google.com/. (Last Accessed: 07 September 2024).

Hartkopf, 2024. Ral: Guidelines for road alignment in germany. https://www.fgsv-verlag.de/pub/media/pdf/201.i.pdf. (Last Accessed: 07 September 2024).

Hirt, C., 2015. *Digital Terrain Models*. Springer, Berlin, Heidelberg.

Jeremy Mennis, R. V., Tomlin, C. D., 2005. Cubic Map Algebra Functions for Spatio-Temporal Analysis. *Cartography and Geographic Information Science*, 32(1), 17–32.

Kumar, P., McElhinney, C. P., Lewis, P., McCarthy, T., 2014. Automated road markings extraction from mobile laser scanning data. *International Journal of Applied Earth Observation and Geoinformation*, 32, 125-137.

Li, Z., Zhu, Q., Gold, C., 2005. *Digital terrain modeling: principles and methodology*. CRC Press, Boca Raton, FL.

Ma, L., Li, Y., Li, J., Wang, C., Wang, R., Chapman, M. A., 2018. Mobile Laser Scanned Point-Clouds for Road Object Detection and Extraction: A Review. *Remote Sensing*, 10(10).

Manandhar, D., Shibasaki, R., 2002. Auto-extraction of urban features from vehicle-borne laser data. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(4), 650–655.

NavVis, 2024. Navvis vlx. https://www.navvis.com/. (Last accessed: 26 March 2024).

Pan, Y., Wang, M., Lu, L., Wei, R., Cavazzi, S., Peck, M., Brilakis, I., 2024. Scan-to-graph: Automatic generation and representation of highway geometric digital twins from point cloud data. *Automation in Construction*, 166, 105654.

Robert, D., Raguet, H., Landrieu, L., 2023. Efficient 3d semantic segmentation with superpoint transformer. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17195–17204.

Schwab, B., Kolbe, T. H., 2022. Validation of Parametirc OpenDRIVE Road Space Models. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W2-2022, 257–264.

Tachikawa, T., Hato, M., Kaku, M., Iwasaki, A., 2011. Characteristics of aster gdem version 2. *2011 IEEE international geoscience and remote sensing symposium*, IEEE, 3657–3660.

Trimble, 2024. Trimble mx9. https://geospatial.trimble.com/en/links?dcs=Collection125839. (Last accessed: 26 March 2024).

Wang, H., Wu, Y., Han, X., Xu, M., Chen, W., 2021. Automatic generation of large-scale 3D road networks based on GIS data. *Computers & Graphics*, 96, 71–81.

Wang, Z., Zhang, H., Lu, Q., Lian, Y., 2018. 3d scene modeling and rendering algorithm based on road extraction of simple attribute remote sensing image. *2018 International Conference on Computer Modeling, Simulation and Algorithm (CMSA 2018)*, Atlantis Press, 12–15.

Williams, K., Olsen, M. J., Roe, G. V., Glennie, C., 2013. Synthesis of Transportation Applications of Mobile LIDAR. *Remote Sensing*, 5(9), 4652–4692.

Zai, D., Li, J., Guo, Y., Cheng, M., Lin, Y., Luo, H., Wang, C., 2017. 3-D road boundary extraction from mobile laser scanning data via supervoxels and graph cuts. *IEEE Transactions on Intelligent Transportation Systems*, 19(3), 802–813.

Zhang, X., Zhong, M., Liu, S., Zheng, L., Chen, Y., 2019. Template-Based 3D Road Modeling for Generating Large-Scale Virtual Road Network Environment. *ISPRS International Journal of Geo-Information*, 8(9).