Urban Building Height Reconstruction from LiDAR, DEM, and Shapefile Building Contours

A Data-Driven Approach for Automated 3D Building Box Modeling Using Python

Jurgis Zagorskas, Daiva Makutėnienė
Department of Engineering Graphics
Vilnius Gediminas Technical University
Vilnius, Lithuania
jurgis.zagorskas@vilniustech.lt

***Abstract*—Accurate and scalable representation of urban environments is essential for urban planning, energy modeling, and geospatial analysis. This study presents a data-driven method for generating simplified 3D building models using building footprints from a shapefile (SHP), digital elevation models (DEM), and filtered LiDAR point clouds. Building heights are estimated by combining DEM-derived ground elevation with maximum LiDAR surface returns within each footprint, and the resulting height difference is used to extrude each footprint into a rectangular 3D building box. The workflow is implemented in Python and includes data preprocessing, height estimation, and 3D geometry generation. Unlike detailed mesh reconstruction methods, the approach prioritizes scalability and computational efficiency for large urban areas, while effectively integrating LiDAR and DEM data to produce lightweight 3D building representations suitable for urban analysis and related applications.**

Keywords—LiDAR point clouds; urban 3D reconstruction; environmental modelling; GIS; Python geoprocessing

I. INTRODUCTION

In recent years, numerous global and regional 3D building datasets have become available for GIS and urban analysis applications. These datasets include open-source building footprint layers such as OpenStreetMap (OSM), as well as commercial and visualization-oriented platforms such as Google Earth 3D Tiles. In addition, several research-driven products and CityGML-based models provide Level of Detail (LOD1–LOD3) building representations for selected regions [1]. However, despite their broad availability, these datasets often suffer from important limitations. Building footprints may be outdated or inconsistent with current urban development and cadastral data, while building height attributes are frequently missing or estimated using indirect methods, leading to significant inaccuracies in practical applications [2,3].

Studies have shown that OpenStreetMap-based 3D building models typically rely on extruded footprints with either missing or approximated height values, which limits their reliability for quantitative analysis such as energy modeling or urban simulation [4,5]. Similarly, global fusion-based datasets and raster-derived height products often provide only coarse height ranges rather than precise building-level measurements, reducing their suitability for detailed urban studies [6]. As a result, there is a growing need for more accurate and scalable methods that derive building heights directly from primary geospatial observations.

One of the most effective data sources for this purpose is LiDAR (Light Detection and Ranging), which is now widely available through national mapping agencies and open geospatial data portals in many countries. LiDAR provides dense 3D point cloud information that enables direct estimation of surface and terrain elevations, making it particularly suitable for precise building height extraction [7]. Previous studies have demonstrated that combining LiDAR data with building footprints enables reliable Level of Detail 1 (LOD1) building reconstruction through prismatic models with flat roofs, which are widely used in urban-scale simulations and spatial analysis applications [6,8].

Beyond height estimation, extensive research has explored more advanced 3D building reconstruction techniques, including roof shape modeling and semantic geometry extraction from LiDAR and multi-source data fusion [9,10]. For example, machine learning and procedural modeling approaches have been proposed to improve building geometry detail and classification accuracy in urban environments [11]. However, despite these advances, such methods remain computationally expensive and are still primarily in the research stage. Although GIS platforms such as ArcGIS Pro and QGIS support 3D visualization and basic extrusion workflows, they do not yet provide fully efficient and automated large-scale pipelines for accurate LiDAR-based building reconstruction.

Therefore, there remains a clear need for efficient, scalable, and reproducible methods that combine building footprint data with LiDAR-derived elevation information and digital elevation models (DEM) to generate accurate and computationally lightweight 3D building representations suitable for large-area urban analysis.

II. LITERATURE REVIEW

Three-dimensional (3D) urban reconstruction has become popular topic in Geographic Information Systems (GIS), remote sensing, and urban analytics due to its importance in planning, energy modeling, and smart city applications [12]. In particular, Level of Detail 1 (LOD1) building models, which represent buildings as simple extruded blocks, are widely used because they provide a computationally efficient abstraction of urban environments while preserving essential geometric information [13]. LOD1 models are typically generated by combining building footprints with height information derived from external data sources such as digital elevation models (DEM) or LiDAR point clouds [14].

A common approach in literature is the fusion of building footprint datasets with elevation information extracted from multi-source remote sensing data [15]. Some examples demonstrate that LOD1 models can be generated by integrating OpenStreetMap (OSM) building footprints with height information derived from remote sensing datasets such as DEMs and SAR-based products, highlighting the general workflow of footprint extrusion based on external height attributes [16]. Similarly, large-scale automated pipelines have been developed to reconstruct millions of buildings using LiDAR-derived point clouds combined with 2D building polygons [17], showing that scalable LOD1 reconstruction is feasible at national scale when reliable elevation data is available.

LiDAR (Light Detection and Ranging) data has been widely recognized as one of the most accurate sources for deriving building heights due to its ability to capture high-resolution 3D surface information [18]. Recent studies show that building height estimation can be effectively performed using statistical measures of LiDAR point distributions within building footprints, such as maximum, median, or percentile-based values [15]. However, the quality of resulting 3D models strongly depends on preprocessing steps such as point classification, filtering of ground and non-ground returns, and accurate footprint segmentation [19].

In addition to height estimation, more advanced research has focused on reconstructing detailed building geometries, including roof structures and semantic building components [20,21]. Deep learning-based methods applied to LiDAR point clouds and aerial imagery have demonstrated significant improvements in footprint extraction and 3D reconstruction accuracy [22]. However, these approaches often require high computational resources and carefully trained models, limiting their applicability for large-scale or real-time processing.

Despite advances in both LiDAR-based reconstruction and machine learning methods, practical GIS tools still offer limited automation for large-scale, fully integrated 3D reconstruction workflows. While software such as ArcGIS Pro and QGIS provide functionalities for visualization and partial 3D extrusion, they do not yet offer fully optimized pipelines that seamlessly integrate LiDAR-derived height extraction, DEM correction, and building footprint processing in an automated manner.

Therefore, existing literature consistently highlights a trade-off between model complexity and scalability. While highly detailed reconstruction methods improve geometric realism, they remain computationally expensive and not suitable for large-scale applications. Conversely, LOD1 approaches offer scalability but depend heavily on the quality and consistency of input datasets. This gap motivates the development of efficient, data-driven methods that combine LiDAR, DEM, and vector building footprints to produce accurate yet computationally lightweight 3D urban models.

III. PROBLEM FORMULATION

Despite the increasing availability of geospatial datasets, accurate and scalable reconstruction of 3D urban building models remains a challenging task. Existing global and regional 3D building datasets typically rely on generalized building footprints and incomplete or estimated height attributes, which often do not reflect real-world urban conditions. As a result, these datasets are insufficient for applications requiring precise building geometry, such as energy modeling, urban microclimate simulation, or detailed spatial analysis.

The core problem addressed in this study is the reliable estimation of building heights and the generation of consistent 3D building representations using heterogeneous geospatial data sources. Specifically, the challenge lies in integrating building footprint polygons (SHP), digital elevation models (DEM), and LiDAR point cloud data into a unified workflow that produces accurate and computationally efficient 3D building models.

A key difficulty arises from inconsistencies between datasets. Building footprints may be outdated or misaligned with actual urban structures, DEM data represents generalized ground elevation rather than localized terrain variations, and LiDAR point clouds require filtering and spatial association to correctly isolate building returns. Therefore, a robust method is required to ensure spatial alignment and consistent extraction of ground and roof elevations within each building footprint.

This study formulates the problem as a per-building height estimation task, where the objective is to compute the vertical extent of each building by combining ground elevation derived from DEM with maximum surface elevation extracted from LiDAR data. The resulting height values are then used to generate simplified 3D building geometries through footprint extrusion.

Formally, given a set of building footprints $F = \{f_1, f_2, \ldots, f_n\}$, a DEM surface $D(x, y)$, and a LiDAR point cloud $L(x, y, z)$, the goal is to compute for each footprint f_i:

- ground elevation g_i from DEM within the footprint area
- roof elevation r_i from filtered LiDAR points within the same footprint
- building height $h_i = r_i - g_i$

The final objective is to transform each footprint f_i into a 3D rectangular solid with height h_i, producing a scalable LOD1 representation suitable for large-area urban analysis.

This formulation emphasizes both accuracy and computational efficiency, enabling processing of large urban datasets while maintaining a balance between geometric simplicity and real-world representativeness.

IV. METHODOLOGICAL WORKFLOW

The overall methodology of the proposed approach is summarized in Fig. 1, which presents the data processing pipeline for generating 3D building models from heterogeneous geospatial datasets. The workflow is structured into three main stages: input data acquisition, processing steps, and output generation.

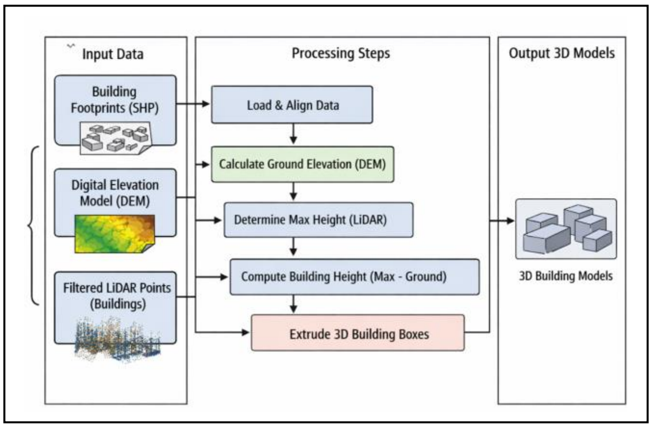

Fig. 1. A flow diagram of proposed building height extraction process pipeline.

In the first stage, three primary data sources are used: building footprints from a shapefile (SHP), a digital elevation model (DEM), and a filtered LiDAR point cloud containing only building-related points. These datasets form the basis for all subsequent computations and must be spatially consistent in terms of coordinate reference systems and alignment.

The second stage consists of sequential processing steps. Initially, all datasets are loaded and spatially aligned to ensure accurate overlay. Ground elevation values are then extracted from the DEM within each building footprint. In parallel, LiDAR points are spatially filtered per footprint to determine the maximum elevation corresponding to building roofs. The building height is computed as the difference between the LiDAR-derived maximum elevation and the DEM-derived ground level. Finally, each building footprint is extruded using the calculated height to generate a simplified 3D box representation.

The final stage produces the output in the form of 3D building models, where each building is represented as a prismatic volume (LOD1). This structured workflow ensures a clear transformation from raw geospatial data to usable 3D urban models, while maintaining computational efficiency and scalability for large-area processing.

The flow diagram highlights the logical sequence of operations and the interaction between datasets, providing a transparent and reproducible framework for implementing the proposed method.

A. *Digital Terrain Model (DEM) Generation From Lidar Data*

A critical component of the proposed methodology is the derivation of an accurate Digital Elevation Model (DEM), representing the ground surface, from raw LiDAR point cloud data. Since LiDAR datasets are typically distributed as tiled .las or .laz files, often compressed into archives, an efficient preprocessing workflow is required to convert these data into raster-based elevation models suitable for further analysis.

The implemented approach automates DEM generation using the PDAL (Point Data Abstraction Library) framework in combination with Python scripting. Initially, LiDAR data are provided as compressed .zip archives containing multiple tiles. Each archive is processed sequentially, and individual LiDAR tiles are extracted into a temporary working directory to avoid excessive disk usage.

For each extracted tile, a PDAL pipeline is executed to convert the point cloud data into a raster DEM (GeoTIFF format). The pipeline typically includes ground point classification filtering (if not pre-classified), followed by interpolation of ground points into a continuous raster surface. Only points classified as ground (e.g., ASPRS class 2) are used to ensure that buildings, vegetation, and other above-ground objects do not influence the terrain model. The rasterization step is performed using GDAL writers within PDAL, producing a georeferenced TIFF file for each tile.

The process is repeated iteratively for all tiles within each archive, and the resulting DEM tiles are stored in a structured directory. This tile-based approach significantly improves computational efficiency and scalability, allowing large LiDAR datasets to be processed without requiring full dataset loading into memory.

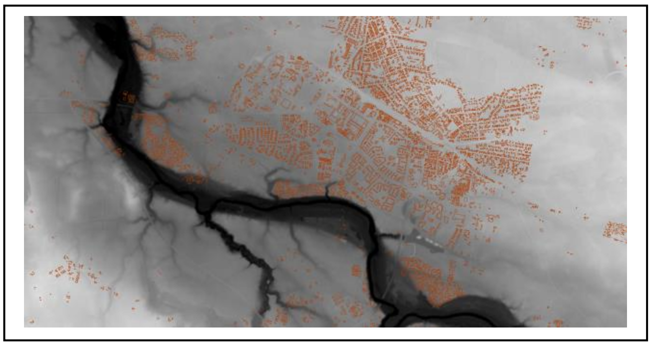

Fig. 2. DEM surface TIFF and building contour geometry SHP.

The resulting DEM represents a high-resolution approximation of the bare-earth surface and serves as the reference for ground elevation in subsequent building height calculations. By deriving the DEM directly from LiDAR data, consistency between terrain and building height measurements is ensured, minimizing vertical discrepancies that may arise when using external or lower-resolution elevation datasets.

Overall, this automated pipeline enables efficient transformation of raw LiDAR point clouds into analysis-ready DEM tiles, forming a reliable foundation for accurate building height estimation and 3D reconstruction.

V. BUILDING HEIGHT EXTRACTION FROM LIDAR POINT CLOUDS

The estimation of building heights is performed directly from LiDAR point cloud data by computing minimum and maximum elevation values within each building footprint. This approach enables accurate height determination without reliance on interpolated raster surfaces, preserving the original precision of LiDAR measurements.

A. Data Preparation

The method requires two primary inputs: (1) building footprint polygons stored in a shapefile, and (2) a LiDAR point cloud (.laz) containing points classified as buildings. The data are loaded using efficient Python libraries:

```
las = laspy.read(laz_path)
x = np.array(las.x, dtype=float)
y = np.array(las.y, dtype=float)
z = np.array(las.z, dtype=float)
```

The LiDAR points are represented as coordinate arrays, enabling vectorized spatial operations.

B. Spatial Indexing with KD-Tree

To ensure computational efficiency, especially for large point clouds, a spatial index is constructed using a k-dimensional tree (KD-tree):

```
points = np.column_stack((x, y))
tree = cKDTree(points)
```

This structure allows rapid querying of nearby points for each building footprint, avoiding the need to iterate through the entire dataset.

C. Candidate Point Selection

For each building polygon, a small buffer is applied to account for minor geometric misalignments between LiDAR data and footprint boundaries:

```
buffered = poly.buffer(buffer_dist)
```

A subset of candidate LiDAR points is then selected using a radius-based query centered on the polygon's bounding box:

```
candidates_idx =
tree.query_ball_point([(minx+maxx)/2,
(miny+maxy)/2],r=radius)
```

This step significantly reduces the number of points that need to be evaluated in subsequent operations.

D. Point-in-Polygon Filtering

The candidate points are further filtered using a vectorized point-in-polygon test to retain only those points that fall within the buffered building footprint:

```
mask_inside = contains_xy(buffered,
x_sub, y_sub)
```

This ensures that only LiDAR points corresponding to the actual building are considered.

E. Height Calculation

For each building, the minimum and maximum elevation values are extracted from the filtered LiDAR points:

```
z_inside = z_sub[mask_inside]
z_min_list.append(z_inside.min())
z_max_list.append(z_inside.max())
```

- Z_min typically represents the lowest LiDAR return associated with the building (e.g., near ground level or base),
- Z_max corresponds to the highest point on the building (e.g., roof peak).

The building height can then be derived as:

$$H = Z_{\max} - Z_{\min}$$

This direct LiDAR-based calculation avoids errors introduced by raster interpolation and provides a more accurate representation of vertical building extent.

F. Output Generation

The computed elevation attributes are stored back into a new shapefile:

```
props['Z_MIN'] = zmin
props['Z_MAX'] = zmax
```

The output dataset preserves original building geometries while enriching them with height-related attributes, enabling further 3D reconstruction (e.g., extrusion into LOD1 models).

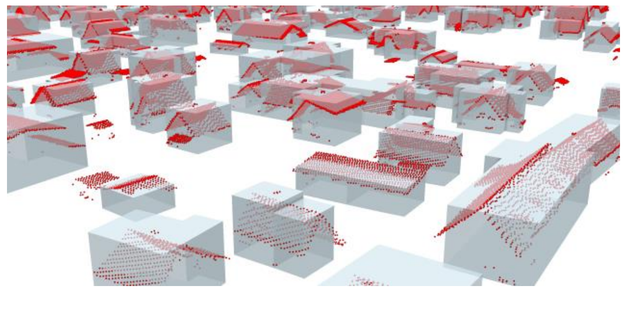

Fig. 3. The resultant boxes created from extrusion of building contour according to the LiDAR data.

Overall, this method provides an efficient and scalable solution for building height extraction by combining spatial indexing, vectorized geometry operations, and direct LiDAR analysis. The use of KD-tree significantly reduces computational complexity, making the approach suitable for large urban datasets while maintaining high accuracy.

Additionally, it should be noted that in cases where buildings have sloped or complex roof structures, the proposed method captures the full vertical extent by relying on the maximum LiDAR elevation within the footprint. As a result, the calculated height represents the highest point of the roof (e.g., ridge or peak), while the minimum value may correspond to lower roof edges or ground-adjacent points. Consequently, the derived height may slightly overestimate the average building height when represented as a simple extruded box (LOD1), since roof geometry is generalized into a flat top (see Figure 3). However, this effect is acceptable within the scope of simplified modeling and does not significantly impact applications that rely on overall building volume or maximum height.

VI. MATHEMATICAL FORMULATION OF KD-TREE–BASED HEIGHT EXTRACTION

The building height estimation problem is formulated as a spatial query and filtering task over a LiDAR point cloud. Let the LiDAR dataset be defined as a set of 3D points:

$$L = \{p_i = (x_i, y_i, z_i) \mid i = 1,2,\dots,N\}$$

where (x_i, y_i) are planar coordinates and z_i is elevation.

Let the set of building footprints be:

$$F = \{f_j \subset \mathbb{R}^2 \mid j = 1,2,\dots,M\}$$

where each $f_j$is a polygon representing a building.

A. KD-Tree Spatial Indexing

To accelerate spatial queries, the LiDAR points are projected onto the horizontal plane and indexed using a KD-tree:

$$P = \{(x_i, y_i)\} \subset \mathbb{R}^2$$

A KD-tree structure T is constructed over P, enabling efficient range queries:

$$T = \text{KDTree}(P)$$

This reduces the complexity of spatial search from $O(N)$per query to approximately $O(\log N)$.

B. Candidate Point Selection

For each building footprint f_j, a buffered polygon $f_j^+$is created to account for spatial misalignment:

$$f_j^+ = \text{buffer}(f_j, d)$$

where $d$is a small buffer distance (e.g., 0.5 m).

A circular query region is defined using the bounding box of f_j^+. Let the center point be:

$$c_j = \left(\frac{x_{\min} + x_{\max}}{2}, \frac{y_{\min} + y_{\max}}{2}\right)$$

and the query radius:

$$r_j = \frac{\max(x_{\max} - x_{\min}, y_{\max} - y_{\min})}{2} + d$$

Candidate points are retrieved using the KD-tree:

$$C_j = \{p_i \in L \mid \| (x_i, y_i) - c_j \| \le r_j\}$$

C. Point-in-Polygon Filtering

From the candidate set C_j, only points located inside the buffered footprint are retained:

$$S_j = \{p_i \in C_j \mid (x_i, y_i) \in f_j^+\}$$

This step ensures geometric correctness by removing points outside the building boundary.

D. Height Estimation

For each building f_j, the minimum and maximum elevations are computed:

$$Z_{\min} = \min_{p_i \in S_j} z_i$$

$$Z_{\max} = \max_{p_i \in S_j} z_i$$

The building height is then defined as:

$$H_j = Z_{\max} - Z_{\min}$$

If no LiDAR points are found within a footprint ($S_j = \emptyset$), the height is treated as undefined.

E. Computational Efficiency

The use of KD-tree indexing significantly reduces the number of points evaluated per building:

- Naïve approach: $O(N \cdot M)$
- KD-tree approach: $O(M \log N + K)$

where $K$is the total number of candidate points processed.

This formulation demonstrates that the proposed method transforms the building height estimation problem into an efficient spatial query and filtering process, combining geometric constraints with optimized data structures to achieve scalable performance on large LiDAR datasets.

This workflow ensures efficient handling of large datasets while maintaining modularity and scalability.

Fig. 4. Building volumes derived from LiDAR point cloud.

A visual example of the proposed methodology is presented in Fig. 4, which illustrates a representative urban district with LiDAR point clouds overlaid by the generated 3D building box models. The figure demonstrates the spatial correspondence between the raw LiDAR data and the resulting simplified building geometries. It can be observed that the extracted building heights closely follow the upper envelope of the LiDAR points, while the footprint-based extrusion produces consistent and coherent volumetric representations.

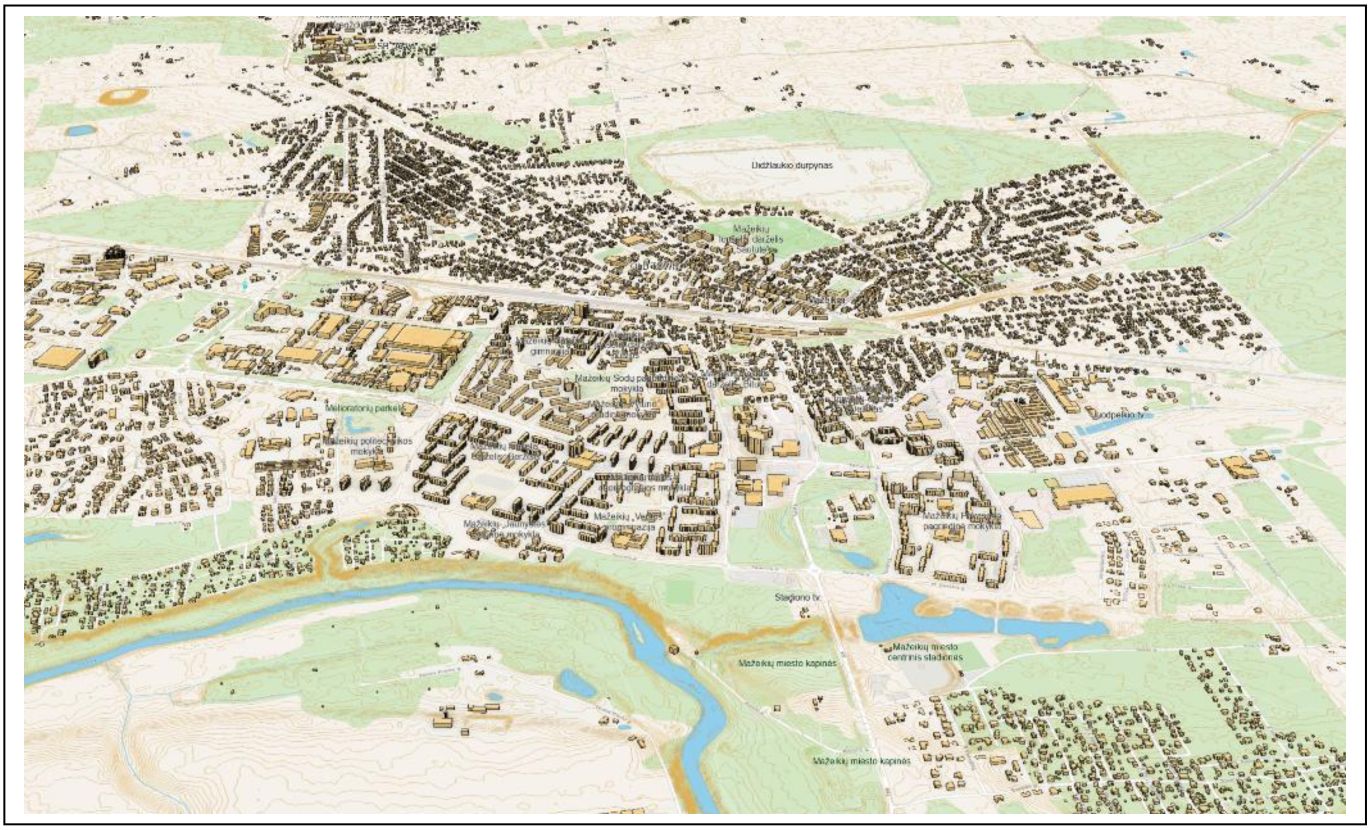

Fig. 5. Building LOD1 representations derived from countour geometry and LiDAR data.

VII. Conclusions

This study presented a data-driven methodology for efficient extraction of building heights and generation of simplified 3D building models using LiDAR point clouds, building footprint polygons, and derived terrain data. The proposed approach integrates vector and point cloud data through spatial indexing and geometric filtering, enabling accurate estimation of building vertical extent based on minimum and maximum LiDAR elevations within each footprint.

The results demonstrate that direct use of LiDAR data provides a reliable alternative to existing global 3D datasets, which often suffer from outdated geometries and generalized or inaccurate height attributes. By avoiding raster interpolation in the height calculation stage and instead operating directly on point clouds, the method preserves the precision of LiDAR measurements while maintaining computational efficiency.

A key contribution of this work is the application of KD-tree–based spatial indexing combined with vectorized point-in-polygon filtering, which significantly reduces computational complexity and allows processing of large urban areas. The use of a small buffer around building footprints further improves robustness against spatial misalignment between datasets.

While more advanced 3D reconstruction methods exist, including detailed roof modeling and semantic segmentation, they remain computationally intensive and are not yet widely applicable for large-scale automated processing. In contrast, the proposed method focuses on Level of Detail 1 (LOD1) representations, providing a practical balance between accuracy and scalability.

The generated building height data and corresponding 3D box models are well suited for applications such as urban density analysis, solar potential assessment, and large-scale GIS simulations. Future work may focus on extending the method to incorporate roof structure detection, improving footprint accuracy, and integrating machine learning techniques for enhanced classification and modeling of urban features.

Overall, the presented approach offers a robust, scalable, and reproducible solution for LiDAR-based urban 3D modeling, addressing current limitations in widely available building datasets.

References

1. Zhu, X.X.; Chen, S.; Zhang, F.; Shi, Y.; Wang, Y. GlobalBuildingAtlas: an open global and complete dataset of building polygons, heights and LoD1 3D models. Earth System Science Data 2025, 17, 6647-6668.
2. Dong, B.; Zheng, Q.; Lin, Y.; Chen, B.; Ye, Z.; Huang, C.; Tong, C.; Li, S.; Deng, J.; Wang, K. Integrating physical model-based features and spatial contextual information to estimate building height in complex urban areas. International Journal of Applied

Earth Observation and Geoinformation 2024, 126, 103625.

3. Stipek, C.; Adams, D.; Dias, P.; Hauser, T.; Lebakula, V.; Sorokine, A.; Epting, J.; Moehl, J.; Stewart, R. A segmented approach to modeling building height: Delineating high-rise and low-rise buildings for enhanced height estimation. Computers, Environment and Urban Systems 2025, 119, 102287.
4. Ullah, T.; Lautenbach, S.; Herfort, B.; Reinmuth, M.; Schorlemmer, D. Assessing completeness of OpenStreetMap building footprints using MapSwipe. ISPRS International Journal of Geo-Information 2023, 12, 143.
5. Mayer, K. Digital Twins at Scale: Accelerating Building Energy Retrofits with Public Data and Machine Learning; Stanford University: 2025.
6. Kaplan, G.; Comert, R.; Kaplan, O.; Matci, D.K.; Avdan, U. Using machine learning to extract building inventory information based on LiDAR data. ISPRS International Journal of Geo-Information 2022, 11, 517.
7. Ma, X.; Zheng, G.; Chi, X.; Yang, L.; Geng, Q.; Li, J.; Qiao, Y. Mapping fine-scale building heights in urban agglomeration with spaceborne lidar. Remote Sensing of Environment 2023, 285, 113392.
8. Huang, X.; Cheng, F.; Bao, Y.; Wang, C.; Wang, J.; Wu, J.; He, J.; Lao, J. Urban building height extraction accommodating various terrain scenes using ICESat-2/ATLAS data. International Journal of Applied Earth Observation and Geoinformation 2024, 130, 103870.
9. Fung, K.Y.; Yang, Z.-L.; Niyogi, D. Improving the local climate zone classification with building height, imperviousness, and machine learning for urban models. Computational Urban Science 2022, 2, 16.
10. Wei, S.; Liu, C.; Tang, N.; Zhao, X.; Zhang, H.; Zhou, X. Indoor and outdoor multi-source 3D data fusion method for ancient buildings. Journal of Measurements in Engineering 2022, 10, 117-139.
11. Nikçi, R.; Župan, R.; Racetin, I. Geovisualization of Buildings: AI vs. Procedural Modeling. Applied Sciences 2024, 14, 8345.
12. Li, Y.; Lai, Y.; Lin, Y. The role of diversified geo-information technologies in urban governance: A literature review. Land 2024, 13, 1408.
13. Breaban, A.-I.; Oniga, V.-E.; Chirila, C.; Loghin, A.-M.; Pfeifer, N.; Macovei, M.; Nicuta Precul, A.-M. Proposed methodology for accuracy improvement of LOD1 3D building models created based on stereo Pléiades satellite imagery. Remote Sensing 2022, 14, 6293.
14. Lakshmanan, L. Integrating LiDAR Point Cloud and Imagery Based Building Footprint for 3D LOD Building Reconstruction. Florida Atlantic University, 2025.
15. Ma, C.; Zhang, Y.; Guo, J.; Zhou, G.; Geng, X. FusionHeightNet: A multi-level cross-fusion method from multi-source remote sensing images for urban building height estimation. Remote Sensing 2024, 16, 958.
16. Zhang, Z.; Xu, Z.; Cao, Y.; Xu, N.; Wang, S.; Cui, S.A.; Li, Z.; Qin, R. Deep learning for 3D point cloud processing--from approaches, tasks to its implications on urban and environmental applications. arXiv preprint arXiv:2509.12452 2025.
17. Xue, B.; Song, Y.; Ai, P.; Li, H.; Liu, S.; Guo, L. Large-Scale Airborne LiDAR Point Cloud Building Extraction Based on Improved Voxelized Deep Learning Network. Buildings 2026, 16, 1450.
18. Chen, H.; Liu, W.; Chen, Q.; Cui, L.; Xing, M. Fine-Grained 3D Building Reconstruction and Floor Height Estimation from Ultra-High-Resolution TomoSAR Data Using Geometric Constraints. Remote Sensing 2026, 18, 1073.
19. AbdulJabbar Sadeq, H. Assessing the effectiveness of filtering algorithms and semantic segmentation for photogrammetric UAV-derived digital terrain models in urban and sloped environments. Survey Review 2025, 1-15.
20. Sun, X.; Guo, B.; Li, C.; Sun, N.; Wang, Y.; Yao, Y. Semantic segmentation and roof reconstruction of urban buildings based on LiDAR point clouds. ISPRS International Journal of Geo-Information 2024, 13, 19.
21. Xiao, X.; Wang, K.; Zhong, Z.; Qu, W.; Wu, W.; Cui, Z.; Su, Y.; Li, A.; Gong, J.; Li, D. A novel data-driven based high-precision building roof contour full-automatic extraction and structured 3D reconstruction method combining stereo images and LiDAR points. International Journal of Digital Earth 2025, 18, 2484668.
22. Buyukdemircioglu, M.; Kocaman, S.; Kada, M. Deep learning for 3D building reconstruction: A review. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2022, 43, 359-366.