TU Delft

Delft University of Technology

City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds

Huang, J.; Stoter, J.E.; Peters, R.Y.; Nan, L.

DOI
10.3390/rs14092254

Publication date
2022

Document Version
Final published version

Published in
Remote Sensing

Citation (APA)
Huang, J., Stoter, J. E., Peters, R. Y., & Nan, L. (2022). City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds. *Remote Sensing*, *14*(9), Article 2254. https://doi.org/10.3390/rs14092254

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

remote sensing

MDPI

Article

City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds

Jin Huang, Jantien Stoter, Ravi Peters and Liangliang Nan *

3D Geoinformation Research Group, Faculty of Architecture and the Built Environment, Delft University of Technology, 2628 BL Delft, The Netherlands; j.huang-1@tudelft.nl (J.H.); j.e.stoter@tudelft.nl (J.S.); r.y.peters@tudelft.nl (R.P.)

* Correspondence: liangliang.nan@tudelft.nl

Abstract: We present a fully automatic approach for reconstructing compact 3D building models from large-scale airborne point clouds. A major challenge of urban reconstruction from airborne LiDAR point clouds lies in that the vertical walls are typically missing. Based on the observation that urban buildings typically consist of planar roofs connected with vertical walls to the ground, we propose an approach to infer the vertical walls directly from the data. With the planar segments of both roofs and walls, we hypothesize the faces of the building surface, and the final model is obtained by using an extended hypothesis-and-selection-based polygonal surface reconstruction framework. Specifically, we introduce a new energy term to encourage roof preferences and two additional hard constraints into the optimization step to ensure correct topology and enhance detail recovery. Experiments on various large-scale airborne LiDAR point clouds have demonstrated that the method is superior to the state-of-the-art methods in terms of reconstruction accuracy and robustness. In addition, we have generated a new dataset with our method consisting of the point clouds and 3D models of 20k real-world buildings. We believe this dataset can stimulate research in urban reconstruction from airborne LiDAR point clouds and the use of 3D city models in urban applications.

Keywords: building reconstruction; LiDAR; point clouds; integer programming

check for updates

Citation: Huang, J.; Stoter, J.; Peters, R.; Nan, L. City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds. *Remote Sens.* **2022**, *14*, 2254. https://doi.org/10.3390/rs14092254

Academic Editors: Mohammad Awrangjeb, Jiaojiao Tian, Qin Yan, Beril Sirmacek and Nusret Demir

Received: 24 March 2022
Accepted: 2 May 2022
Published: 7 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Digitizing urban scenes is an important research problem in computer vision, computer graphics, and photogrammetry communities. Three-dimensional models of urban buildings have become the infrastructure for a variety of real-world applications such as visualization [1], simulation [2–4], navigation [5], and entertainment [6]. These applications typically require high-accuracy and compact 3D building models of large-scale urban environments.

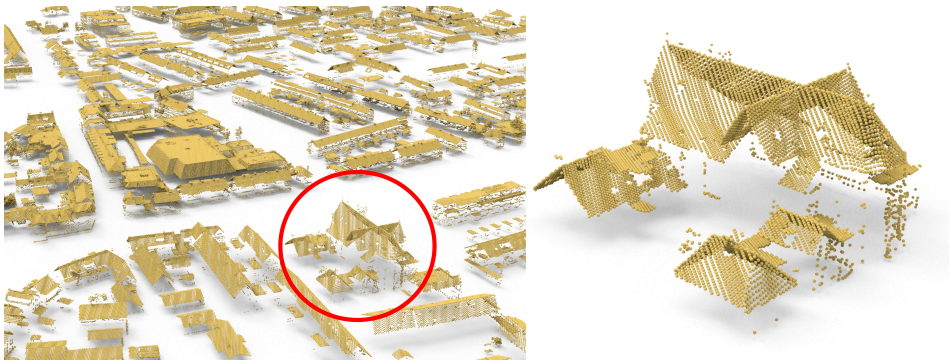
Existing urban building reconstruction methods strive to bring in a great level of detail and automate the process for large-scale urban environments. Interactive reconstruction techniques are successful in reconstructing accurate 3D building models with great detail [7,8], but they require either high-quality laser scans as input or considerable amounts of user interaction. These methods can thus hardly be applied to large-scale urban scenes. To facilitate practical applications that require large-scale 3D building models, researchers have attempted to address the reconstruction challenge using various data sources [9–16]. Existing methods based on aerial images [10,12,13] and dense triangle meshes [11] typically require good coverage of the buildings, which imposes challenges in data acquisition [17]. Approaches based on airborne LiDAR point clouds alleviate data acquisition issues. However, the accuracy and geometric details are usually compromised [9,14–16]. Following previous works using widely available airborne LiDAR point clouds, we strive to recover desired geometric details of real-world buildings while ensuring topological correctness, reconstruction accuracy, and good efficiency.

The challenges for large-scale urban reconstruction from airborne LiDAR point clouds include:

- Building instance segmentation. Urban scenes are populated with diverse objects, such as buildings, trees, city furniture, and dynamic objects (e.g., vehicles and pedestrians). The cluttered nature of urban scenes poses a severe challenge to the identification and separation of individual buildings from the massive point clouds. This has drawn considerable attention in recent years [18,19].
- Incomplete data. Some important structures (e.g., vertical walls) of buildings are typically not captured in airborne LiDAR point clouds due to the restricted positioning and moving trajectories of airborne scanners.
- Complex structures. Real-world buildings demonstrate complex structures with varying styles. However, limited cues about structure can be extracted from the sparse and noisy point clouds, which further introduces ambiguities in obtaining topologically correct surface models.

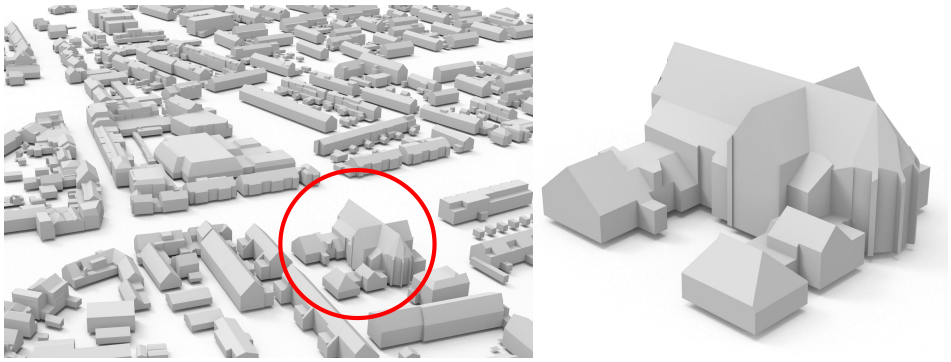
In this work, we address the above challenges with the following strategies. Firstly, we address the building instance segmentation challenge by separating individual buildings using increasingly-available vectorized building footprint data. Secondly, we exploit prior knowledge about the structures of buildings to infer their vertical planes. Based on the fact that vertical planes in airborne LiDAR point clouds are typically walls connecting the piecewise planar roofs to the ground, we propose an algorithm to infer the vertical planes from incomplete point clouds. Our method has the option to extrude outer walls directly from the given building footprint. Finally, we approach surface reconstruction by introducing the inferred vertical planes as constraints into an existing hypothesis-and-selection-based polygonal surface reconstruction framework [20], which favors good fitting to the input point cloud, encourages compactness, and enforces manifoldness of the final model (see Figure 1 for an example of the reconstruction results). The main contributions of this work include:

- A robust framework for fully automatic reconstruction of large-scale urban buildings from airborne LiDAR point clouds.
- An extension of an existing hypothesis-and-selection-based surface reconstruction method for buildings, which is achieved by introducing a new energy term to encourage roof preferences and two additional hard constraints to ensure correct topology and enhance detail recovery.
- A novel approach for inferring vertical planes of buildings from airborne LiDAR point clouds, for which we introduce an optimal-transport method to extract polylines from 2D bounding contours.
- A new dataset consisting of the point clouds and reconstructed surface models of 20 k real-world buildings.



(**a**) Input airborne LiDAR point cloud.

Figure 1. *Cont.*



(**b**) Our reconstruction result

Figure 1. The automatic reconstruction result of all the buildings in a large scene from the AHN3 dataset [21].

2. Related Work

A large volume of methods for urban building reconstruction has been proposed. In this section, we mainly review the techniques relevant to the key components of our method. Since our method relies on footprint data for extracting building instances from the massive point clouds of large scenes, and it can also be used for footprint extraction, we also discuss related techniques in footprint extraction.

Roof primitive extraction. The commonly used method for extracting basic primitives (e.g., planes and cylinders) from point clouds is random sample consensus (RANSAC) [22] and its variants [23,24], which are robust against noise and outliers. Another group of widely used methods is based on region growing [25–27], which assumes roofs are piecewise planar and iteratively propagates planar regions by advancing the boundaries. The main difference between existing region growing methods lies in the generation of seed points and the criteria for region expansion. In this paper, we utilize an existing region growing method to extract roof primitives given its simplicity and robustness, which is detailed in Rabbani et al. [25].

Footprint extraction. Footprints are 2D outlines of buildings, capturing the geometry of outer walls projected onto the ground plane. Methods for footprint extraction commonly project the points to a 2D grid and analyze their distributions [28]. Chen et al. [27] detect rooftop boundaries and cluster them by taking into account topological consistency between the contours. To obtain simplified footprints, polyline simplification methods such as the Douglas-Peucker algorithm [29] are commonly used to reduce the complexity of the extracted contours [12,30,31]. To favor structural regularities, Zhou and Neumann [32] compute the principal directions of a building and regularize the roof boundary polylines along with these directions. Following these works, we infer the vertical planes of a building by detecting its contours from a heightmap generated from a 2D projection of the input points. The contour polylines are then regularized by orientation-based clustering followed by an adjustment step.

Building surface reconstruction. This type of methods aims at obtaining a simplified surface representation of buildings by exploiting geometric cues, e.g., planar primitives and their boundaries [15,32–36]. Zhou and Neumann [37] approached this by simplifying the 2.5D TIN (triangulated irregular network) of buildings, which may result in artifacts in building contours due to its limited capability in capturing complex topology. To address this issue, the authors proposed an extended 2.5D contouring method with improved topology control [38]. To cope with missing walls, Chauve et al. [39] also incorporated additional primitives inferred from the point clouds. Another group of building surface reconstruction methods involves predefined building parts, commonly known as model-driven approaches [40,41]. These methods rely on templates of known roof structures and deform-to-fit the templates to the input points. Therefore, the results are usually limited to the predefined shape templates, regardless of the diverse and complex nature of roof structures or high intraclass variations. Given the fact that buildings demonstrate mainly

piecewise planar regions, methods have also been proposed to obtain an arrangement of extracted planar primitives to represent the building geometry [20,42–44]. These methods first detect a set of planar primitives from the input point clouds and then hypothesize a set of polyhedral cells or polygonal faces using the supporting planes of the extracted planar primitives. Finally, a compact polygonal mesh is extracted from the hypothesized cells or faces. These methods focus on the assembly of planar primitives, for which obtaining a complete set of planar primitives from airborne LiDAR point clouds is still a challenge.

In this work, we extend an existing hypothesis-and-selection-based general polygonal surface reconstruction method [20] to reconstruct buildings that consist of piecewise planar roofs connected to the ground by vertical walls. We approach this by introducing a novel energy term and a few hard constraints specially designed for buildings to ensure correct topology and decent details.

3. Methodology

3.1. Overview

The proposed approach takes as input a raw airborne LiDAR point cloud of a large urban scene and the corresponding building footprints, and it outputs 2-manifold and watertight 3D polygonal models of the buildings in the scene. Figure 2 shows the pipeline of the proposed method. It first extracts the point clouds of individual buildings by projecting all points onto the ground plane and collecting the points lying inside the footprint polygon of each building. Then, we reconstruct a compact polygonal model from the point cloud of each building.

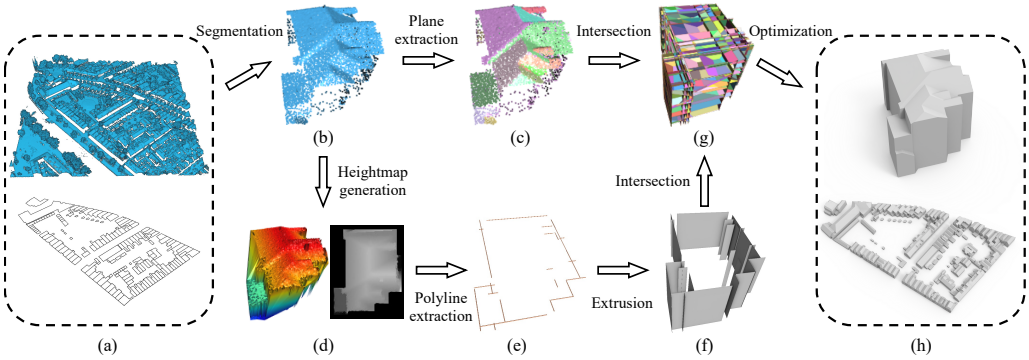


Figure 2. The pipeline of the proposed method (only one building is selected to illustrate the workflow). (**a**) Input point cloud and corresponding footprint data. (**b**) A building extracted from the input point cloud using its footprint polygon. (**c**) Planar segments extracted from the point cloud. (**d**) The heightmap (right) generated from the TIN (left, colored as a height field). (**e**) The polylines extracted from the heightmap. (**f**) The vertical planes obtained by extruding the inferred polylines. (**g**) The hypothesized building faces generated using both the extracted planes and inferred vertical planes. (**h**) The final model obtained through optimization.

Our reconstruction of a single building is based on the hypothesis-and-selection-based framework of PolyFit [20], which is for reconstructing general piecewise-planar objects from a set of planar segments extracted from the point cloud. Our method exploits not only the planar segments directly extracted from the point cloud but also the vertical planes inferred from the point cloud. From these two types of planar primitives, we hypothesize the faces of the building. The final model is then obtained by choosing the optimal subset of the faces through optimization.

The differences between our method and PolyFit are: (1) our method is dedicated to reconstructing urban buildings, and it makes use of vertical planes as hard constraints, for which we propose a novel algorithm for inferring the vertical planes of buildings that are commonly missing in airborne LiDAR point clouds. (2) We introduce a new *roof preference* energy term and two additional hard constraints into the optimization to ensure correct

topology and enhance detail recovery. In the following sections, we detail the key steps of our method with an emphasis on the processes that differ from PolyFit [20].

3.2. Inferring Vertical Planes

With airborne LiDAR point clouds, important structures like vertical walls of a building are commonly missed due to the restricted positioning and moving trajectories of the scanner. In contrast, the roof surfaces are usually well captured. This inspired us to infer the missing walls from the available points containing the roof surfaces. We infer the vertical planes representing not only the outer walls but also the vertical walls within the footprint of a building. We achieve this by generating a 2D rasterized height map from its 3D points and looking for the contours that demonstrate considerable variations in the height values. To this end, an optimal-transport method is proposed to extract closed polylines from the contours. The polylines are then extruded to obtain the vertical walls. The process for inferring the vertical planes is outlined in Figure 2d–f.

Specifically, after obtaining the point cloud of a building, we project the points onto the ground plane, from which we create a height map. To cope with the non-uniform distribution of the points (e.g., some regions have holes while others may have repeating points), we construct a Triangulated Irregular Network (TIN) model using 2D Delaunay triangulation. The TIN model is a continuous surface and naturally completes the missing regions. Then, a height map is generated by rasterizing the TIN model with a specified resolution r. The issue of small holes in the height maps (due to uneven distribution of roof points) is further alleviated by image morphological operators while preserving the shape and size of the building [45]. After that, a set of contours are extracted from the height map using the Canny detector [46], which serves as the initial estimation of the vertical planes. We propose an optimal-transport method to extract polylines from the initial set of contours.

Optimal-transport method for polyline extraction. The initial set of contours are discrete pixels, denoted as S, from which we would like to extract simplified polylines that best describe the 2D geometry of S. Our optimal-transport method for extracting polylines from S works as follows. First, a 2D Delaunay triangulation T_0 is constructed from the discrete points in S. Then, the initial triangulation T_0 is simplified through iterative edge collapse and vertex removal operations. In each iteration, the most suitable vertex to be removed is determined in a way such that the following conditions are met:

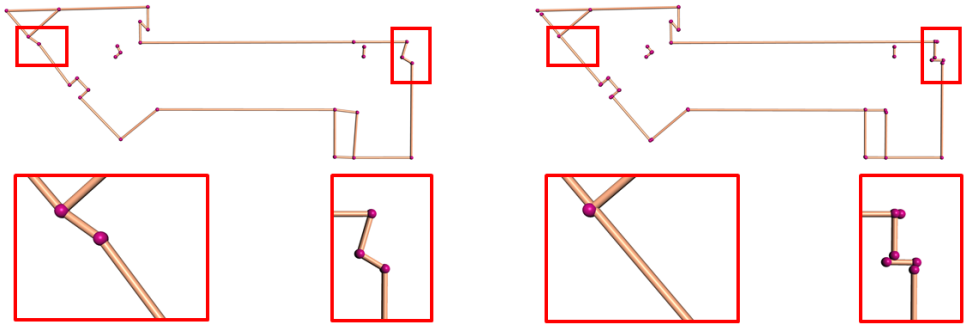
- The maximum Hausdorff distance from the simplified mesh T_0 to S is less than a distance threshold ϵ_d.
- The increase of the total transport cost [47] between S and T_0 is kept at a minimum.

In each iteration, a vertex satisfying the above conditions is removed from T_0 by edge collapse, and the overall transportation cost is updated.

As the iterative simplification process continues, the overall transportation cost will increase. The edge collapse operation stops until no vertex can be further removed, or the overall transportation cost has increased above a user-specified tolerance ϵ_c. After that, we apply an edge filtering step [47] to eliminate small groups of undesirable edges caused by noise and outliers. Finally, the polylines are derived from the remaining vertices and edges of the simplified triangulation using the procedure described in [47]. Compared to [47], our method not only minimizes the total transport cost but also provides control over local geometry, ensuring that the distance between every vertex in the final polylines and the initial contours is smaller than the specified distance threshold ϵ_d.

Regularity enhancement. Due to noise and uneven point density in the point cloud, the polylines generated by the optimal-transport algorithm are unavoidably inaccurate and irregular (see Figure 3a), which often leads to artifacts in the final reconstruction. We alleviate these artifacts by enforcing structure regularities that commonly dominate urban buildings. We consider the structure regularities, namely parallelism, collinearity, and orthogonality, defined by [48]. Please note that since all the lines will be extruded vertically to obtain the vertical planes, the verticality regularity will inherently be satisfied. We

propose a clustering-based method to identify the groups of line segments that potentially satisfy these regularities. Our method achieves structure regularization in two steps: clustering and adjustment.



(**a**) Before (28 segments) (**b**) After (22 segments)

Figure 3. The effect of the clustering-based regularity enhancement on the polylines inferring the vertical walls. (**a**) Before regularity enhancement. (**b**) After regularity enhancement.

Clustering. In this work, we cluster the line segments of the polylines generated by the optimal-transport algorithm based on their orientation and pairwise Euclidean distance [49]. The pairwise Euclidean distance is measured by the minimum distance between a line segment and the supporting line of the other line segment.

Adjustment. For each cluster that contains multiple line segments, we compute its average direction. Then each line segment in the cluster is adjusted to align with the average direction. In case the building footprint is provided, the structure regularity can be further improved by aligning the segments with the edges in the footprint. After average adjustment, the near-collinear and near-orthogonal line segments are adjusted to be perfectly collinear and orthogonal, respectively (we use an angle threshold of 20°).

After regularity enhancement, the vertical planes of the building can be obtained by vertical extrusion of the regularized polylines. The effect of the regularity enhancement is demonstrated in Figure 3, from which we can see that it significantly improves structure regularity and reduces the complexity of the building outlines.

3.3. Reconstruction

Our surface reconstruction involves two types of planar primitives, i.e., vertical planes inferred in the previous step (see Section 3.2) and roof planes directly extracted from the point cloud. Unlike PolyFit [20] that hypothesizes faces by computing pairwise intersections using all planar primitives, we compute pairwise intersections using only the roof planes, and then the resulted faces are cropped with the outer vertical planes (see Figure 2g). This process ensures that the roof boundaries of the reconstructed building can be precisely connected with the inferred vertical walls. Additionally, since the object to be reconstructed is a real-world building, we introduce a *roof preference* energy term and a set of new hard constraints specially designed for buildings into the original formulation. Specifically, our objective for obtaining the model faces F^* can be written as

$$F^* = \arg\min_X \lambda_d E_d + \lambda_c E_c + \lambda_r E_r, \tag{1}$$

where $X = \{x_i | x_i \in \{0, 1\}\}$ denotes the binary variables for the faces (1 for *selected* and 0 otherwise). E_d is the data fitting term that encourages selecting faces supported by more points, and E_c is the model complexity term that favors simple planar structures. For more details about the data fitting term and the model complexity term, please refer to the original paper of PolyFit [20]. In the following part, we elaborate on the new energy term and hard constraints.

New energy term: *roof preference*. We have observed in rare cases that a building in aerial point clouds may demonstrate more than one layer of roofs, e.g., semi-transparent or overhung roofs. In such a case, we assume a higher roof face is always preferable to the ones underneath. We formulate this preference as an additional energy term called *roof preference*, which is defined as

$$E_r = \frac{1}{|F|} \sum_{i=1}^{|F|} x_i \cdot \frac{z_{max} - z_i}{z_{max} - z_{min}} \tag{2}$$

where z_i denotes the Z coordinate of the centroid of a hypothsized face f_i. z_{max} and z_{min} are, respectively, the highest and lowest Z coordinates of the building points. $|F|$ denotes the total number of hypothesized faces.

New hard constraints. We impose two hard constraints to enhance the topological correctness of the final reconstruction.

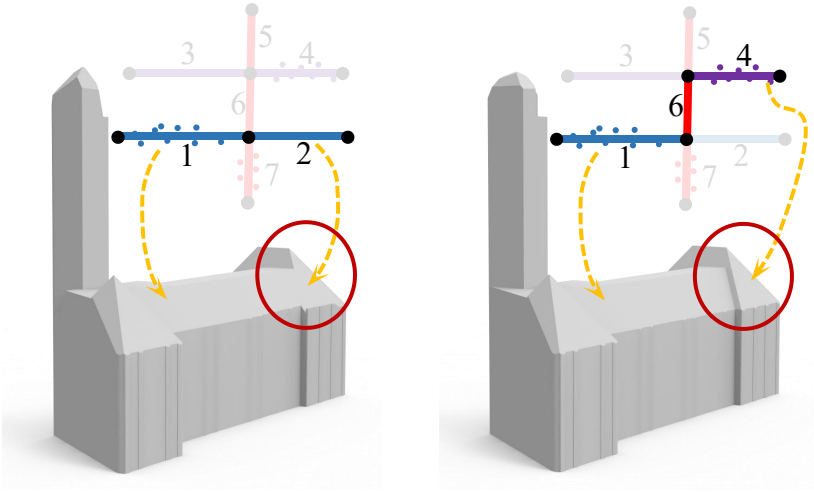
- *Single-layer roof*. This constraint ensures that the reconstructed 3D model of a real-world building has a single layer of roofs, which can be written as,

$$\sum_{k \in V(f_i)} x_k = 1, (1 \leq i \leq |F|)$$

  where $V(f_i)$ denotes the set of hypothesized faces that have overlap with face $f_i \in F$ in the vertical direction.
- *Face prior*. This constraint enforces that for all the derived faces from the same planar segment, the one with the highest confidence value is always selected as a prior. Here, the confidence of a face is measured by the number of its supporting points. This constraint can be simply written as

$$x_l = 1,$$

  where x_l is the variable whose value denotes the status of the most confident face f_l of a planar segment. This constraint resolves ambiguities if two hypothesized faces are near coplanar and close to each other, which preserves finer geometric details. The effect of this constraint is demonstrated in Figure 4.



(**a**) (**b**)

Figure 4. The effect of the *face prior* constraint. The insets illustrate the assembly of the hypothesized faces in the corresponding marked regions (each line segment denotes a hypothesized face, and line segments of the same color represent faces derived from the same planar primitive). (**a**) Reconstruction without the *face prior* constraint. (**b**) Reconstruction with the *face prior* constraint, for which faces 1 and 4 both satisfy the *face prior* constraint. The numbers 1–7 denote the 7 candidate faces.

The final surface model of the building can be obtained by solving the optimization problem given in Equation (A4), subject to the *single-layer roof* and *face prior* hard constraints.

4. Results and Evaluation

Our method is implemented in C++ using CGAL [50]. All experiments were conducted on a desktop PC with a 3.5 GHz AMD Ryzen Threadripper 1920X and 64 GB RAM.

4.1. Test Datasets

We have tested our method on three datasets of large-scale urban point clouds including more than 20 k buildings.

- AHN3 [21]. An openly available country-wide airborne LiDAR point cloud dataset covering the entire Netherlands, with an average point density of 8 points/m^2. The corresponding footprints of the buildings are obtained from the Register of Buildings and Addresses (BAG) [51]. The geometry of footprint is acquired from aerial photos and terrestrial measurements with an accuracy of 0.3 m. The polygons in the BAG represent the outlines of buildings as their outer walls seen from above, which are slightly different from footprints. We still use 'footprint' in this paper.
- DALES [52]. A large-scale aerial point cloud dataset consisting of forty scenes spanning an area of 10 km^2, with instance labels of 6 k buildings. The data was collected using a Riegl Q1560 dual-channel system with a flight altitude of 1300 m above ground and a speed of 72 m/s. Each area was collected by a minimum of 5 laser pulses per meter in four directions. The LiDAR swaths were calibrated using the BayesStripAlign 2.0 software and registered, taking both relative and absolute errors into account and correcting for altitude and positional errors. The average point density is 50 points/m^2. No footprint data is available in this dataset.
- Vaihingen [53]. An airborne LiDAR point cloud dataset published by ISPRS, which has been widely used in semantic segmentation and reconstruction of urban scenes. The data were obtained using a Leica ALS50 system with 45° field of view and a mean flying height above ground of 500 m. The average strip overlap is 30% and multiple pulses were recorded. The point cloud was pre-processed to compensate for systematic offsets between the strips. We use in our experiments a training set that contains footprint information and covers an area of 399 m $\times$ 421 m with 753 k points. The average point density is 4 points/m^2.

4.2. Reconstruction Results

Visual results. We have used our method to reconstruct more than 20 k buildings from the aforementioned three datasets. For the AHN3 [21] and Vaihingen [53] datasets, the provided footprints were used for both building instance segmentation and extrusion of the outer walls. Our inferred vertical planes were used to complete the missed inner walls. For the DALES [52] dataset, we used the provided instance labels to extract building instances, and we used our inferred vertical walls for the reconstruction.

Figures 1 and 5 show the 3D reconstruction of all buildings in two large scenes from the AHN3 dataset [21]. For the buildings reconstructed in Figure 1, their models are simplified polygonal meshes with an average face count of 34. To better reveal the quality of our reconstructed building models, we demonstrate in Figure 6 a set of individual buildings reconstructed from the three test datasets. From these visual results, we can see that although the buildings have diverse structures of different styles, and the input point clouds have varying densities and different levels of noise, outliers, and missing data, our method succeeded in obtaining visually plausible reconstruction results. These experiments also indicate that our approach is successful in inferring the vertical planes of buildings from airborne LiDAR point clouds and it is effective to include these planes in the 3D reconstruction of urban buildings.



Figure 5. Reconstruction of a large scene from the AHN3 dataset [21].

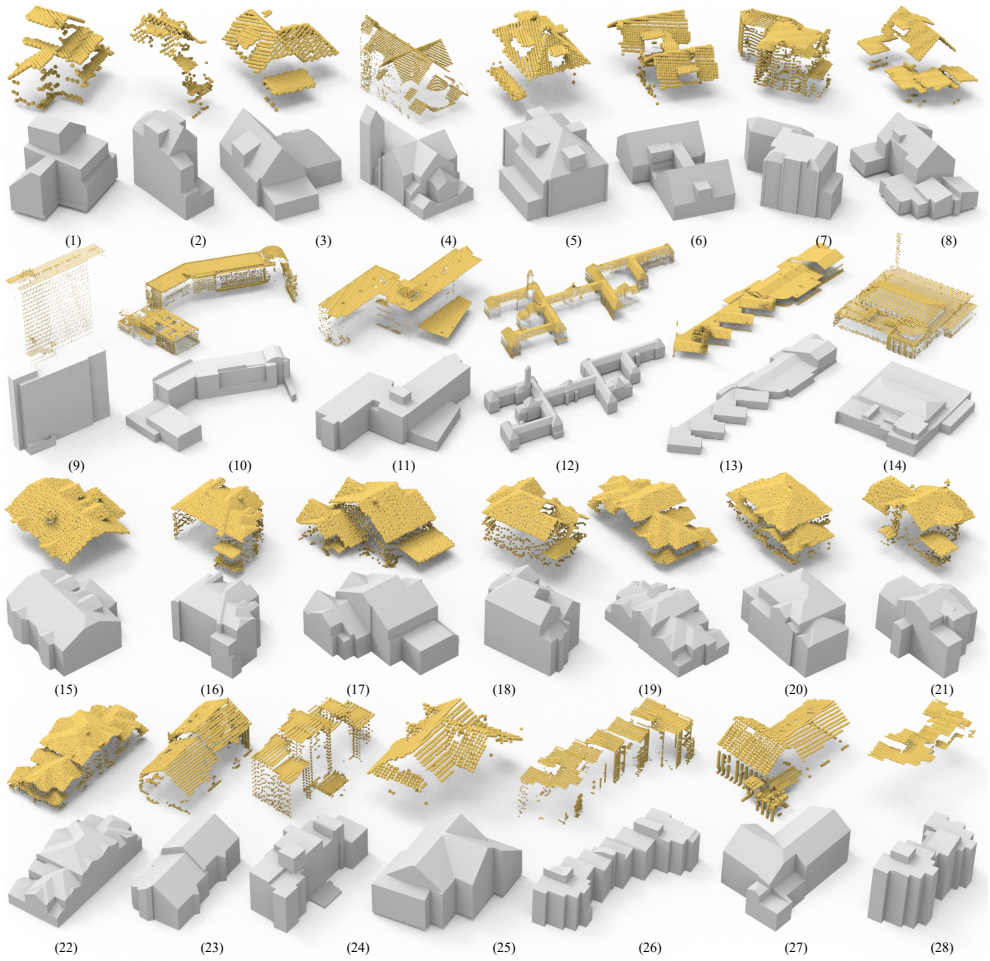


Figure 6. The reconstruction results of a set of buildings from various dataset. (**1–14**) are from the AHN3 dataset [21], (**15–22**) are from the DALES dataset [52], (**23–28**) are from the Vaihingen dataset [53].

Quantitative results. We have also evaluated the reconstruction results quantitatively. Since ground-truth reconstruction is not available for all buildings in the three datasets, we

chose to use the commonly used accuracy measure, Root Mean Square Error (RMSE), to quantify the quality of each reconstructed model. In the context of surface reconstruction, RMSE is defined as the square root of the average of squared Euclidean distances from the points to the reconstructed model. In Table 1, we report the statistics of our quantitative results on the buildings shown in Figure 6. We can see that our method has obtained good reconstruction accuracy, i.e., the RMSE for all buildings is between 0.04 m to 0.26 m, which is quite promising for 3D reconstruction of real-world buildings from noisy and sparse airborne LiDAR point clouds. As observed from the number of faces column of Table 1, our results are simplified polygonal models and are more compact than those obtained from commonly used approaches such as the Poisson surface reconstruction method [54] (that produces dense triangles). Table 1 also shows that the running times for most buildings are less than 30 s. The reconstruction of the large complex building shown in Figure 6 (12) took 42 min. This long reconstruction time is due to that our method computes the pairwise intersection of the detected planar primitives and inferred vertical planes, and it generates a large number of candidate faces and results in a large optimization problem [20] (see also Section 4.7). The running time with respect to the number of detected planar segments for the reconstruction of more buildings is reported in Figure 7.

Table 1. Statistics on the reconstructed buildings shown in Figure 6. For each building, the number of points in the input, number of faces in the reconstructed model, fitting error (i.e., RMSE in meters), and running time (in seconds) are reported.

Dataset	Model	#Points	#Faces	RMSE (m)	Time (s)
AHN3	(1)	732	23	0.07	3
	(2)	532	42	0.12	4
	(3)	1165	31	0.04	3
	(4)	20,365	127	0.15	62
	(5)	1371	48	0.04	5
	(6)	1611	45	0.06	4
	(7)	3636	68	0.21	18
	(8)	2545	52	0.04	8
	(9)	15,022	63	0.11	28
	(10)	23,654	262	0.26	115
	(11)	13,269	102	0.11	34
	(12)	155,360	1520	0.09	2520
	(13)	24,027	176	0.24	141
	(14)	28,522	227	0.15	78
DALES	(15)	8662	39	0.04	11
	(16)	11,830	73	0.1	8
	(17)	10,673	47	0.07	7
	(18)	7594	33	0.07	14
	(19)	13,060	278	0.05	145
	(20)	11,114	55	0.06	24
	(21)	8589	51	0.06	15
	(22)	18,909	282	0.08	86

Table 1. *Cont.*

Dataset	Model	#Points	#Faces	RMSE (m)	Time (s)
Vaihingen	(23)	7701	51	0.24	25
	(24)	6845	99	0.12	8
	(25)	1007	24	0.11	2
	(26)	11,591	206	0.17	10
	(27)	4026	42	0.26	6
	(28)	5059	61	0.22	9

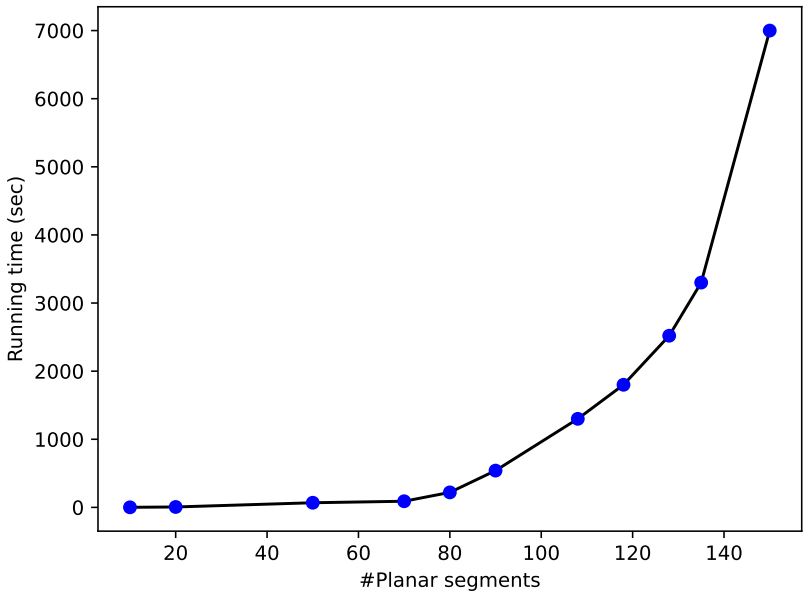


Figure 7. The running time of our method with respect to the number of the detected planar segments. These statistics are obtained by testing on the AHN3 dataset.

New dataset. Our method has been applied to city-scale building reconstruction. The results are released as a new dataset consisting of 20 k buildings (including the reconstructed 3D models and the corresponding airborne LiDAR point clouds). We believe this dataset can stimulate research in urban reconstruction from airborne LiDAR point clouds and the use of 3D city models in urban applications.

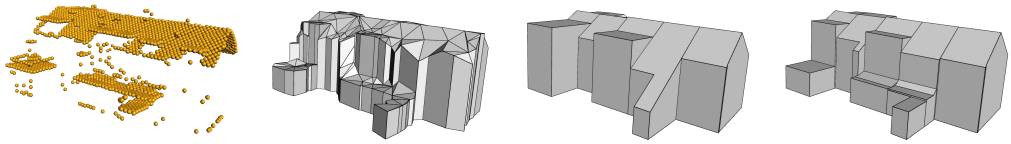
4.3. Parameters

Our method involves a few parameters that are empirically set to fixed values for all experiments, i.e., the distance threshold $\epsilon_d = 0.25$ and the tolerance for overall transportation cost $\epsilon_c = 2.0$. The resolution r for the rasterization of the TIN model to generate heightmaps is dataset dependent due to the difference in point density. It is set to 0.20 m from AHN3, 0.15 m for DALES, and 0.25 m for Vaihingen. The weight of the *roof preference* energy term $\lambda_r = 0.04$ (while the weights for the data fitting and model complexity terms are set to $\lambda_d = 0.34$ and $\lambda_c = 0.62$, respectively).

4.4. Comparisons

We have compared our method with two successful open-source methods, i.e., 2.5D Dual Contouring (dedicated for urban buildings) [37] and PolyFit (for general piecewise-planar objects) [20], on the AHN3 [21], DALES [52], and Vaihingen [53] datasets. The city block from the AHN3 dataset [21] is sparse and contains only 80,447 points for 160 buildings (i.e., on average 503 points per building). The city region from DALES is denser and contains

214,601 points for 41 buildings (i.e., on average 5234 points per building). The city area from the Vaihingen dataset contains 69,254 points for 57 buildings (i.e., on average 1215 points per building). The walls of all the point clouds are severely occluded. Figure 8 shows the visual comparison of one of the buildings. PolyFit assumes a complete set of input planar primitives, which is not the case for airborne LiDAR point clouds because the vertical walls are often missing. For PolyFit to be effective, we added our inferred vertical planes to its initial set of planar primitives. From the result, we can observe that both PolyFit and our method can generate compact building models, and the number of faces in the result is an order of magnitude less than that of the 2.5D Dual Contouring method. It is worth noting that even with the additional planes, PolyFit still failed to reconstruct some walls and performed poorly in recovering geometric details. In contrast, our method produces the most plausible 3D models. By inferring missing vertical planes, our method can recover inner walls, which further split the roof planes and bring in more geometric details into the final reconstruction. Table 2 reports the statistics of the comparison, from which we can see that the reconstructed building models from our method have the highest accuracy. In terms of running time, our method is slower than the other two, but it is still acceptable in practical applications (on average 4.9 s per building).



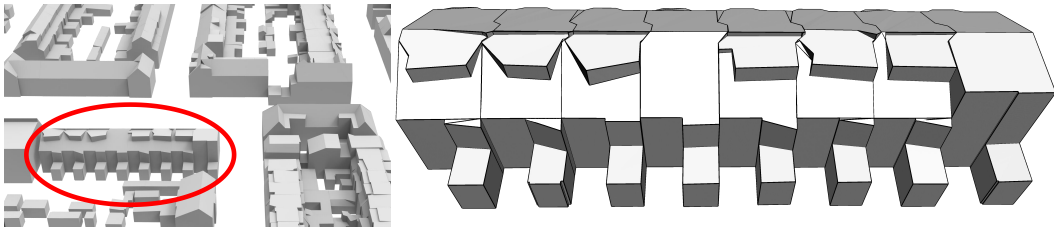
(**a**) Input point cloud (**b**) 2.5DC (296 faces) (**c**) PolyFit (58 faces) (**d**) Ours (86 faces)

Figure 8. Comparison with 2.5D Dual Contouring (2.5DC) [37] and PolyFit [20] on a single building from the AHN3 dataset [21].

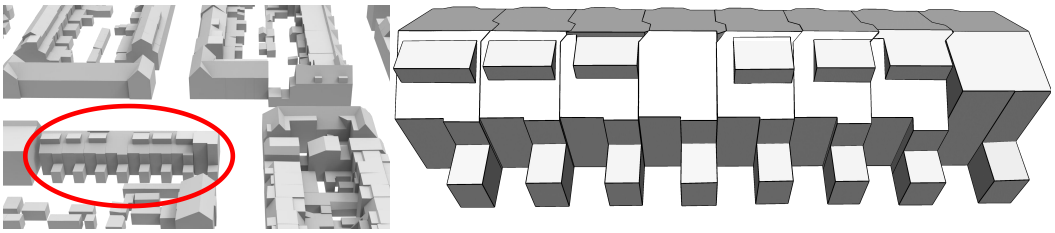
Table 2. Statistics on the comparison of 2.5D Dual Contouring [37], PolyFit [20], and our method on the reconstruction from the AHN3 [21], DALES [52], and Vaihingen [53] datasets. Total face numbers, running times, and average errors are reported.

Dataset	Method	#Faces	RMSE (m)	Time (s)
AHN3	2.5D DC [37]	12,781	0.213	13
	PolyFit [20]	1848	0.242	160
	Ours	2453	0.128	380
DALES	2.5D DC [37]	2297	0.204	10
	PolyFit [20]	444	0.287	230
	Ours	583	0.184	670
Vaihingen	2.5D DC [37]	2695	0.168	6
	PolyFit [20]	647	0.275	102
	Ours	798	0.157	212

We also performed an extensive quantitative comparison with the 3D building models from the BAG3D [55], which is a public 3D city platform that provides 3D models of urban buildings at the LoD2 level. For this comparison, we picked four different regions consisting of 1113 buildings in total from the BAG3D. In Figure 9, we demonstrate a visual comparison, from which we can see that our models demonstrate more regularity. The quantitive result is reported in Table 3, from which we can see that our results have higher accuracy.



(**a**) Result from BAG3D



(**b**) Ours result

Figure 9. A visual comparison with BAG3D [55]. A building from Table 3 (b) is shown.

Table 3. Quantitative comparison with the BAG3D [55] on four urban scenes (a)–(d). Both BAG3D and our method used the point clouds from the AHN3 dataset [21] as input. The bold font indicates smaller RMSE values.

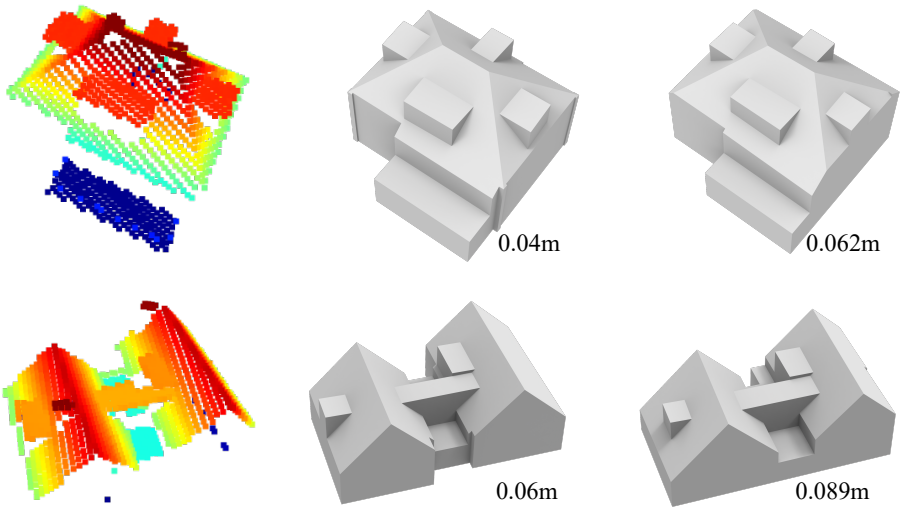
Region	#Points	#Building	RMSE (m) BAG3D	RMSE (m) Ours
(a)	1,694,247	198	0.088	**0.079**
(b)	329,593	387	0.139	**0.138**
(c)	224,970	368	0.140	**0.132**
(d)	80,447	160	0.146	**0.128**

4.5. With vs. Without Footprint

Our method can infer the vertical planes of a building from its roof points, and then the outer walls are completed using the vertical planes. It also has the option to directly use given footprint data for reconstruction. With a given footprint, vertically planes are firstly obtained by extruding the footprint polygons. Then these planes and those extracted from the point clouds are intersected to hypothesize the model faces, followed by the optimization step to obtain the final reconstruction. Figure 10 shows such a comparison on two buildings.

4.6. Reconstruction Using Point Clouds with Vertical Planes

The methodology presented in our paper only focuses on airborne LiDAR point clouds, in which vertical walls of buildings are typically missing. In practice, our method can be easily adapted to work with other types of point clouds that contain points of vertical walls, e.g., point clouds reconstructed from drone images. For such point clouds, our method can still be effective by replacing the inferred vertical planes with those directly detected from the point clouds. Figure 11 shows two such examples.



(a) Input (b) With footprint (c) Without footprint

Figure 10. Comparison between the reconstruction *with* (**b**) and *without* (**c**) footprint data on two buildings (**a**) from the AHN3 dataset [21]. The number below each model denotes the root mean square error (RMSE). Using the inferred vertical planes slightly increases reconstruction errors.

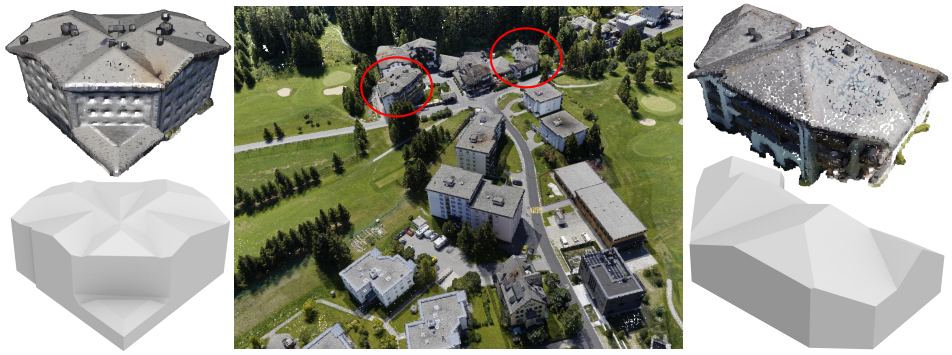
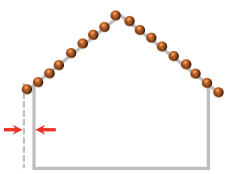


Figure 11. Reconstruction from aerial point clouds. In these point clouds, the vertical walls can be extracted from the point clouds and directly used in reconstruction, and thus the vertical plane inference step was skipped. The dataset is obtained from Can et al. [56].

4.7. Limitations

Our method can infer the missing vertical planes of buildings, from which the outer vertical planes serve as outer walls in the reconstruction. Since the vertical planes are inferred from the 3D points of rooftops, the walls in the final models may not perfectly align with the ground-truth footprints (see the figure below). Thus, we recommend the use of high-quality footprint data whenever it is available. Besides, our method extends the hypothesis-and-selection-based surface reconstruction framework of PolyFit [20] by introducing new energy terms and hard constraints. It naturally inherits the limitation of PolyFit, i.e., it may encounter computation bottlenecks for buildings with complex structures (e.g., buildings with more than 100 planar regions). An example has already been shown in Figure 6 (12).



5. Conclusions and Future Work

We have presented a fully automatic approach for large-scale 3D reconstruction of urban buildings from airborne LiDAR point clouds. We propose to infer the vertical planes of buildings that are commonly missing from airborne LiDAR point clouds. The inferred vertical planes play two different roles during the reconstruction. The outer vertical planes directly become part of the exterior walls of the building, and the inner vertical planes enrich building details by splitting the roof planes at proper locations and forming the necessary inner walls in final models. Our method can also incorporate given building footprints for reconstruction. In case footprints are used, they are extruded to serve the exterior walls of the models, and the inferred inner planes enrich building details. Extensive experiments on different datasets have demonstrated that inferring vertical planes is an effective strategy for building reconstruction from airborne LiDAR point clouds, and the proposed *roof preference* energy term and the novel hard constraints ensure topologically correct and accurate reconstruction.

Our current framework uses only planar primitives and it is sufficient for reconstructing most urban buildings. In the real world, there still exist buildings with curved surfaces, which our current implementation could not handle. However, our hypothesize-and-selection strategy is general and can be extended to process different types of primitives. As a future work direction, our method can be extended to incorporate other geometric primitives, such as spheres, cylinders, or even parametric surfaces. With such an extension, buildings with curved surfaces can also be reconstructed.

Author Contributions: J.H. performed the study and implemented the algorithms. R.P. and J.S. provided constructive comments and suggestions. L.N. proposed this topic, provided daily supervision, and wrote the paper together with J.H. All authors have read and agreed to the published version of the manuscript.

Funding: Jin Huang is financially supported by the China Scholarship Council.

Data Availability Statement: Our code and data are available at https://github.com/yidahuang/City3D, accessed on 23 March 2022.

Acknowledgments: We thank Zexin Yang, Zhaiyu Chen, and Noortje van der Horst for proofreading the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LiDAR	Light Detection and Ranging
TIN	Triangular Irregular Network
RMSE	Root Mean Square Error

Appendix A. The Complete Formulation

Our reconstruction is obtained by finding the optimal subset of the hypothesized faces. We formulate this as an optimization problem, with an objective function consisting of three energy terms: *data fitting*, *model complexity*, and *roof preference*. The first two terms are the same as in [20]. In the following, we briefly introduce all these terms and provide the final complete formulation.

- **Data fitting**. It is defined to measure how well the final model (i.e., the assembly of the chosen faces) fits to the input point cloud,

$$E_d = 1 - \frac{1}{|P|} \sum_{i=1}^{|F|} x_i \cdot support(f_i), \tag{A1}$$

where $|P|$ is the number of points in the point cloud. $support(f_i)$ measures the number of points that are ϵ-close to a face $f_i \in F$, and $x_i \in \{0,1\}$ denotes the binary status of the face f_i (1 for *selected* and 0 otherwise). $|F|$ denotes the total number of hypothesized faces.

- **Model complexity**. To avoid defects introduced by noise and outliers, this term is introduced to encourage large planar structures,

$$E_c = \frac{1}{|E|} \sum_{i=1}^{|E|} corner(e_i), \tag{A2}$$

where $|E|$ denotes the total number of pairwise intersections in the hypothesized face set. $corner(e_i)$ is an indicator function denoting if choosing two faces connected by an edge e_i results in a sharp edge in the final model (1 for *sharp* and 0 otherwise).

- **Roof preference**. We have observed in rare cases that a building in aerial point clouds may demonstrate more than one layer of roofs, e.g., semi-transparent or overhung roofs. In such a case, we assume a higher roof face is preferable to the ones underneath. We formulate this preference as an additional *roof preference* energy term,

$$E_r = \frac{1}{|F|} \sum_{i=1}^{|F|} x_i \cdot \frac{z_{max} - z_i}{z_{max} - z_{min}} \tag{A3}$$

where z_i denotes the Z coordinate of the centroid of a face f_i. z_{max} and z_{min} are, respectively, the highest and lowest Z coordinates of the building points.

With all the constraints, the complete optimization problem is written as

$$\begin{aligned} &\min_X \lambda_d E_d + \lambda_c E_c + \lambda_r E_r \\ &\text{s.t.} \quad \begin{cases} \sum\limits_{k \in V(f_i)} x_k = 1, (1 \le i \le |F|) \\ \sum\limits_{j \in N(e_i)} x_j = 0 \quad or \quad 2, \quad (1 \le j \le |E|) \\ x_l = 1, \\ x_i \in \{0,1\}, \quad \forall i \in N \end{cases} \end{aligned} \tag{A4}$$

where the first constraint is call *single roof*, which ensures that the reconstructed building model has a single layer of roofs. The second constraint enforces that in the final model an edge is associated with two adjacent faces, ensuring the final model to be watertight and manifold. The third constraint is call *face prior*, which ensures that, for the faces derived from the same planar segment, the one with the highest confidence value is selected as a prior.

By solving the above optimization problem, the set of selected faces $\{f_i | x_i = 1\}$ forms the final surface model of a building.

References

1. Yao, Z.; Nagel, C.; Kunde, F.; Hudra, G.; Willkomm, P.; Donaubauer, A.; Adolphi, T.; Kolbe, T.H. 3DCityDB—A 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. *Open Geospat. Data Softw. Stand.* **2018**, *3*, 1–26. [CrossRef]
2. Zhivov, A.M.; Case, M.P.; Jank, R.; Eicker, U.; Booth, S. Planning tools to simulate and optimize neighborhood energy systems. In *Green Defense Technology*; Springer: Dordrecht, The Netherlands, 2017; pp. 137–163.
3. Stoter, J.; Peters, R.; Commandeur, T.; Dukai, B.; Kumar, K.; Ledoux, H. Automated reconstruction of 3D input data for noise simulation. *Comput. Environ. Urban Syst.* **2020**, *80*, 101424. [CrossRef]
4. Widl, E.; Agugiaro, G.; Peters-Anders, J. Linking Semantic 3D City Models with Domain-Specific Simulation Tools for the Planning and Validation of Energy Applications at District Level. *Sustainability* **2021**, *13*, 8782. [CrossRef]
5. Cappelle, C.; El Najjar, M.E.; Charpillet, F.; Pomorski, D. Virtual 3D city model for navigation in urban areas. *J. Intell. Robot. Syst.* **2012**, *66*, 377–399. [CrossRef]

6. Kargas, A.; Loumos, G.; Varoutas, D. Using different ways of 3D reconstruction of historical cities for gaming purposes: The case study of Nafplio. *Heritage* **2019**, *2*, 1799–1811. [CrossRef]
7. Nan, L.; Sharf, A.; Zhang, H.; Cohen-Or, D.; Chen, B. Smartboxes for interactive urban reconstruction. In *ACM Siggraph 2010 Papers*; ACM: New Yrok, NY, USA, 2010; pp. 1–10.
8. Nan, L.; Jiang, C.; Ghanem, B.; Wonka, P. Template assembly for detailed urban reconstruction. In *Computer Graphics Forum*; Wiley Online Library: Zurich, Switzerland, 2015; Volume 34, pp. 217–228.
9. Zhou, Q.Y. *3D Urban Modeling from City-Scale Aerial LiDAR Data*; University of Southern California: Los Angeles, CA, USA, 2012.
10. Haala, N.; Rothermel, M.; Cavegn, S. Extracting 3D urban models from oblique aerial images. In Proceedings of the 2015 Joint Urban Remote Sensing Event (JURSE), Lausanne, Switzerland, 30 March–1 April 2015; pp. 1–4.
11. Verdie, Y.; Lafarge, F.; Alliez, P. LOD generation for urban scenes. *ACM Trans. Graph.* **2015**, *34*, 30. [CrossRef]
12. Li, M.; Nan, L.; Smith, N.; Wonka, P. Reconstructing building mass models from UAV images. *Comput. Graph.* **2016**, *54*, 84–93. [CrossRef]
13. Buyukdemircioglu, M.; Kocaman, S.; Isikdag, U. Semi-automatic 3D city model generation from large-format aerial images. *ISPRS Int. J.-Geo-Inf.* **2018**, *7*, 339. [CrossRef]
14. Bauchet, J.P.; Lafarge, F. City Reconstruction from Airborne Lidar: A Computational Geometry Approach. In Proceedings of the 3D GeoInfo 2019—14thConference 3D GeoInfo, Singapore, 26–27 September 2019.
15. Li, M.; Rottensteiner, F.; Heipke, C. Modelling of buildings from aerial LiDAR point clouds using TINs and label maps. *ISPRS J. Photogramm. Remote Sens.* **2019**, *154*, 127–138. [CrossRef]
16. Ledoux, H.; Biljecki, F.; Dukai, B.; Kumar, K.; Peters, R.; Stoter, J.; Commandeur, T. 3dfier: Automatic reconstruction of 3D city models. *J. Open Source Softw.* **2021**, *6*, 2866. [CrossRef]
17. Zhou, X.; Yi, Z.; Liu, Y.; Huang, K.; Huang, H. Survey on path and view planning for UAVs. *Virtual Real. Intell. Hardw.* **2020**, *2*, 56–69. [CrossRef]
18. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
19. Thomas, H.; Qi, C.R.; Deschaud, J.E.; Marcotegui, B.; Goulette, F.; Guibas, L.J. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6411–6420.
20. Nan, L.; Wonka, P. PolyFit: Polygonal Surface Reconstruction from Point Clouds. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October, 2017.
21. AHN3. Actueel Hoogtebestand Nederland (AHN). 2018. Available online: https://www.pdok.nl/nl/ahn3-downloads (accessed on 13 November 2021).
22. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
23. Schnabel, R.; Wahl, R.; Klein, R. Efficient RANSAC for point-cloud shape detection. In *Computer Graphics Forum*; Wiley Online Library: Oxford, UK, 2007; Volume 26, pp. 214–226.
24. Zuliani, M.; Kenney, C.S.; Manjunath, B. The multiransac algorithm and its application to detect planar homographies. In Proceedings of the IEEE International Conference on Image Processing 2005, Genova, Italy, 14 September 2005; Volume 3, p. III-153.
25. Rabbani, T.; Van Den Heuvel, F.; Vosselmann, G. Segmentation of point clouds using smoothness constraint. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2006**, *36*, 248–253.
26. Sun, S.; Salvaggio, C. Aerial 3D building detection and modeling from airborne LiDAR point clouds. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2013**, *6*, 1440–1449. [CrossRef]
27. Chen, D.; Wang, R.; Peethambaran, J. Topologically aware building rooftop reconstruction from airborne laser scanning point clouds. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 7032–7052. [CrossRef]
28. Meng, X.; Wang, L.; Currit, N. Morphology-based building detection from airborne LIDAR data. *Photogramm. Eng. Remote Sens.* **2009**, *75*, 437–442. [CrossRef]
29. Douglas, D.H.; Peucker, T.K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartogr. Int. J. Geogr. Inf. Geovis.* **1973**, *10*, 112–122. [CrossRef]
30. Zhang, K.; Yan, J.; Chen, S.C. Automatic construction of building footprints from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 2523–2533. [CrossRef]
31. Xiong, B.; Elberink, S.O.; Vosselman, G. Footprint map partitioning using airborne laser scanning data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *3*, 241–247. [CrossRef]
32. Zhou, Q.Y.; Neumann, U. Fast and extensible building modeling from airborne LiDAR data. In Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Irvine, CA, USA, 5–7 November 2008; pp. 1–8.
33. Dorninger, P.; Pfeifer, N. A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. *Sensors* **2008**, *8*, 7323–7343. [CrossRef]
34. Lafarge, F.; Mallet, C. Creating large-scale city models from 3D-point clouds: A robust approach with hybrid representation. *Int. J. Comput. Vis.* **2012**, *99*, 69–85. [CrossRef]

35. Xiao, Y.; Wang, C.; Li, J.; Zhang, W.; Xi, X.; Wang, C.; Dong, P. Building segmentation and modeling from airborne LiDAR data. *Int. J. Digit. Earth* **2015**, *8*, 694–709. [CrossRef]
36. Yi, C.; Zhang, Y.; Wu, Q.; Xu, Y.; Remil, O.; Wei, M.; Wang, J. Urban building reconstruction from raw LiDAR point data. *Comput.-Aided Des.* **2017**, *93*, 1–14. [CrossRef]
37. Zhou, Q.Y.; Neumann, U. 2.5 d dual contouring: A robust approach to creating building models from aerial lidar point clouds. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2010; pp. 115–128.
38. Zhou, Q.Y.; Neumann, U. 2.5 D building modeling with topology control. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 2489–2496.
39. Chauve, A.L.; Labatut, P.; Pons, J.P. Robust piecewise-planar 3D reconstruction and completion from large-scale unstructured point data. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 1261–1268.
40. Lafarge, F.; Descombes, X.; Zerubia, J.; Pierrot-Deseilligny, M. Structural approach for building reconstruction from a single DSM. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *32*, 135–147. [CrossRef]
41. Xiong, B.; Elberink, S.O.; Vosselman, G. A graph edit dictionary for correcting errors in roof topology graphs reconstructed from point clouds. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 227–242. [CrossRef]
42. Li, M.; Wonka, P.; Nan, L. Manhattan-world Urban Reconstruction from Point Clouds. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016.
43. Bauchet, J.P.; Lafarge, F. Kinetic shape reconstruction. *ACM Trans. Graph. (TOG)* **2020**, *39*, 1–14. [CrossRef]
44. Fang, H.; Lafarge, F. Connect-and-Slice: An hybrid approach for reconstructing 3D objects. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 13490–13498.
45. Huang, H.; Brenner, C.; Sester, M. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. *ISPRS J. Photogramm. Remote Sens.* **2013**, *79*, 29–43. [CrossRef]
46. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *PAMI-8*, 679–698. [CrossRef]
47. De Goes, F.; Cohen-Steiner, D.; Alliez, P.; Desbrun, M. An optimal transport approach to robust reconstruction and simplification of 2D shapes. In *Computer Graphics Forum*; Wiley Online Library: Oxford, UK, 2011; Volume 30, pp. 1593–1602.
48. Li, Y.; Wu, B. Relation-Constrained 3D Reconstruction of Buildings in Metropolitan Areas from Photogrammetric Point Clouds. *Remote Sens.* **2021**, *13*, 129. [CrossRef]
49. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst. (TODS)* **2017**, *42*, 1–21. [CrossRef]
50. CGAL Library. *CGAL User and Reference Manual*, 5.0.2 ed.; CGAL Editorial Board: Valbonne, French, 2020.
51. BAG. Basisregistratie Adressen en Gebouwen (BAG). 2019. Available online: https://bag.basisregistraties.overheid.nl/datamodel (accessed on 13 November 2021).
52. Varney, N.; Asari, V.K.; Graehling, Q. DALES: A large-scale aerial LiDAR data set for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 186–187.
53. Rottensteiner, F.; Sohn, G.; Jung, J.; Gerke, M.; Baillard, C.; Benitez, S.; Breitkopf, U. The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci. I-3* **2012**, *1*, 293–298. [CrossRef]
54. Kazhdan, M.; Bolitho, M.; Hoppe, H. Poisson surface reconstruction. In Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Cagliari, Italy, 26–28 June 2006; Volume 7.
55. 3D BAG (v21.09.8). 2021. Available online: https://3dbag.nl/en/viewer (accessed on 13 November 2021).
56. Can, G.; Mantegazza, D.; Abbate, G.; Chappuis, S.; Giusti, A. Semantic segmentation on Swiss3DCities: A benchmark study on aerial photogrammetric 3D pointcloud dataset. *Pattern Recognit. Lett.* **2021**, *150*, 108–114. [CrossRef]