

Article

A Robust Automatic Method to Extract Building Facade Maps from 3D Point Cloud Data

Bing Yu ^{1,†}, Jinlong Hu ^{1,†} , Xiujuan Dong ^{1,2,*}, Keren Dai ^{2,3} , Dongsheng Xiao ¹, Bo Zhang ⁴ , Tao Wu ¹, Yunliang Hu ¹ and Bing Wang ¹

¹ School of Civil Engineering and Geomatics, Southwest Petroleum University, Chengdu 610500, China

² State Key Laboratory of Geohazard Prevention and Geoenvironment Protection, Chengdu University of Technology, Chengdu 610059, China

³ College of Earth Science, Chengdu University of Technology, Chengdu 610059, China

⁴ Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China

* Correspondence: dongxiujuan@cdut.cn

† These authors contributed equally to this work.

Abstract: Extracting facade maps from 3D point clouds is a fast and economical way to describe a building's surface structure. Existing methods lack efficiency, robustness, and accuracy, and depend on many additional features such as point cloud reflectivity and color. This paper proposes a robust and automatic method to extract building facade maps. First, an improved 3D Hough transform is proposed by adding shift vote and 3D convolution of the accumulator to improve computational efficiency and reduce peak fuzziness and dependence on the step selection. These modifications make the extraction of potential planes fast and accurate. Second, the coplane and vertical plane constraints are introduced to eliminate pseudoplanes and nonbuilding facades. Then, we propose a strategy to refine the potential facade and to achieve the accurate calibration and division of the adjacent facade boundaries by clustering the refined point clouds of the facade. This process solves the problem where adjoining surfaces are merged into the same surface in the traditional method. Finally, the extracted facade point clouds are converted into feature images. Doors, windows, and building edges are accurately extracted via deep learning and digital image processing techniques, which combine to achieve accurate extraction of building facades. The proposed method was tested on the MLS and TLS point cloud datasets, which were collected from different cities with different building styles. Experimental results confirm that the proposed method decreases computational burden, improves efficiency, and achieves the accurate differentiation of adjacent facade boundaries with higher accuracy compared with the traditional method, verifying the robustness of the method. Additionally, the proposed method uses only point cloud geometry information, effectively reducing data requirements and acquisition costs.

Keywords: facade extraction and refinement; automatic facade map extraction; improved 3D Hough transform; deep learning



Citation: Yu, B.; Hu, J.; Dong, X.; Dai, K.; Xiao, D.; Zhang, B.; Wu, T.; Hu, Y.; Wang, B. A Robust Automatic Method to Extract Building Facade Maps from 3D Point Cloud Data. *Remote Sens.* **2022**, *14*, 3848. <https://doi.org/10.3390/rs14163848>

Academic Editor: Bailang Yu

Received: 7 July 2022

Accepted: 6 August 2022

Published: 9 August 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

Buildings form the dominant artificial objects in urban scenes. The requirements for accurate building geometries and three-dimensional (3D) building models are growing in tandem with the expansion of urban planning, smart city construction, and building information modeling (BIM). How to efficiently and accurately obtain these data and the information required for 3D modeling is a key issue [1]. Building facade maps represent the geometric features of building surfaces, such as the edges of windows, doors, and other vital structures. Facade maps can directly serve urban renewal, urban planning, etc., while providing a flexible and straightforward approach to retrieving large-scale building models [1,2]. Laser scanning provides a quick and accurate method to gather 3D point

cloud data (PCD) from 3D objects [3]. Thus, how to extract the required geometric features from 3D PCD accurately and robustly should be determined.

Multiple methods to extract facade maps from 3D PCDs have been proposed, and direct or indirect extraction is the most common approach [1]. In direct extraction methods, facade maps are obtained directly from raw or processed 3D PCD by computing geometric information. Given that 3D PCD storage is unstructured, building facade maps are typically extracted by random sample consensus (RANSAC) [4], region growing [5], or semantic feature-based approaches [1,6,7]. These algorithms are typically efficient and concise but only apply to specific situations and rely on good data quality. Slicing-based methods are other commonly used direct extraction methods that can effectively extract facade maps using hole and edge detection, and are easy to use [8–10]. However, they are strongly affected by occlusion. In contrast, indirect extraction, which includes segmentation and feature extraction, is a more prevalent approach [11]. Segmentation segregates a group of points into several single surfaces or regions. Building segmentation, which separates various sides of a building, including walls and roofs, from one another, is typically a precursor of feature extraction. Fuzzy clustering [12,13], 3D Hough transform (HT) [14–16], RANSAC [17], and other methods are often used for building segmentation. The fuzzy clustering method has high complexity, and its results depend on the initialization parameters. The RANSAC method has higher accuracy and is less affected by noise but can only match one instance at a time and typically achieves multiple instance acquisition via iterative elimination [17,18]. Thus, its results are strongly influenced by the algorithm parameters and convergence conditions, making the results unstable. The 3D HT method can extract multiple instances from point cloud data at once, but the step size limits its accuracy. Some models suited to 3D PCD were proposed with the prosperity of deep learning, and outstanding achievements in the field of point cloud segmentation have been made [19–21]. However, these models are complex and have strict hardware requirements, while having poor universality to different scenarios and even different data. Feature extraction involves extracting architectural features (e.g., doors and windows) from segmented parts. Commonly used methods include a priori semantic features, slicing, region growth, etc. [11,22], which typically achieve better results when data quality is high, showing low robustness and generalizability.

In general, existing methods typically achieve good results in ideal environments [4], but in practice, due to the occlusion, noise, and the uneven density of PCDs, they still have marked limitations in: (1) Strict data requirements. Some methods (e.g., clustering-based approaches and improved region growth methods [22,23]) rely on a variety of feature information, which makes it difficult to manage PCD containing only coordinate information, raising the hardware requirements and cost of PCD acquisition. Many algorithms also lack robustness and perform poorly when data quality is poor; (2) High manual work requirements. It is challenging to automatically extract building facade maps directly from unordered PCD; (3) Low transferability. Deep-learning-based point cloud segmentation methods are highly automated, but have low transferability due to the point cloud data's unstructured characteristics and unstable data quality. One method or model can work well on specific data but performs poorly on others.

To address these limitations, we propose a new method for automatic and robust building facade map extraction. This method can extract building facade geometry and the edges of windows and doors based only on the coordinate information of the 3D PCD without the assistance of other feature information (e.g., laser intensity, color). The IQmulus & TerraMobilita Contest and Semantic3D.Net Benchmark datasets were used to test the proposed method. The primary contributions of this study are as follows:

(1) A new method is presented to extract building facades from 3D PCD. First, an improved 3D HT algorithm is proposed by adding shift vote and 3D convolution to the 3D HT, which improves the accuracy and efficiency of potential facade extraction. Then, the improved 3D HT and RANSAC are combined to achieve potential facade refinement. Thus, the facade extraction's accuracy and robustness markedly increase compared with

the conventional 3D HT and RANSAC. Additionally, the improved 3D HT method is more robust and has a lower data dependency than the deep-learning-based point cloud segmentation methods, which are data-driven.

(2) A facade boundary calibration method is proposed. Planes in a mathematical sense without a definite range are transformed into real building facades with definite boundaries using a density-based clustering method. This method can distinguish facades from other objects and different facades in proximity, improving the extraction accuracy of building facades and avoiding different facades being mistakenly merged into one.

(3) A new way to extract building facade maps from feature images is proposed. The Faster R-CNN model, a classical deep-learning-based image object detection model, is introduced to extract the door and window edges from the feature images. This method achieves better results despite poor data quality (e.g., presence of occlusion, noise, uneven density) compared to traditional geometry-based methods [24].

2. Methodology

The proposed method includes two steps: building facade extraction and building facade map extraction. A 3D PCD was imported as input, each facade equation and its corresponding range were obtained by building facade extraction, and point cloud division was implemented to accurately identify different building facades. Based on these facade data, the building facade map was obtained by building facade map extraction.

Building facade extraction includes three steps: (1) potential plane acquisition; (2) facade constraint; and (3) facade precise extraction. Building facade map extraction also includes three steps: (1) feature image generation; (2) door and window detection; and (3) building boundary extraction. More details are described in the following subsections.

2.1. Building Facade Extraction

In order to overcome the common defects (i.e., high influence by data factors such as noise, uneven density, occlusion, no clear planar boundary, etc.) in point cloud plane segmentation methods, a new method is proposed to extract building facades that has a high robustness compared to traditional methods. The proposed method's workflow is shown in Figure 1 and consists of three primary steps: potential plane acquisition, facade constraints, and facade precise extraction.

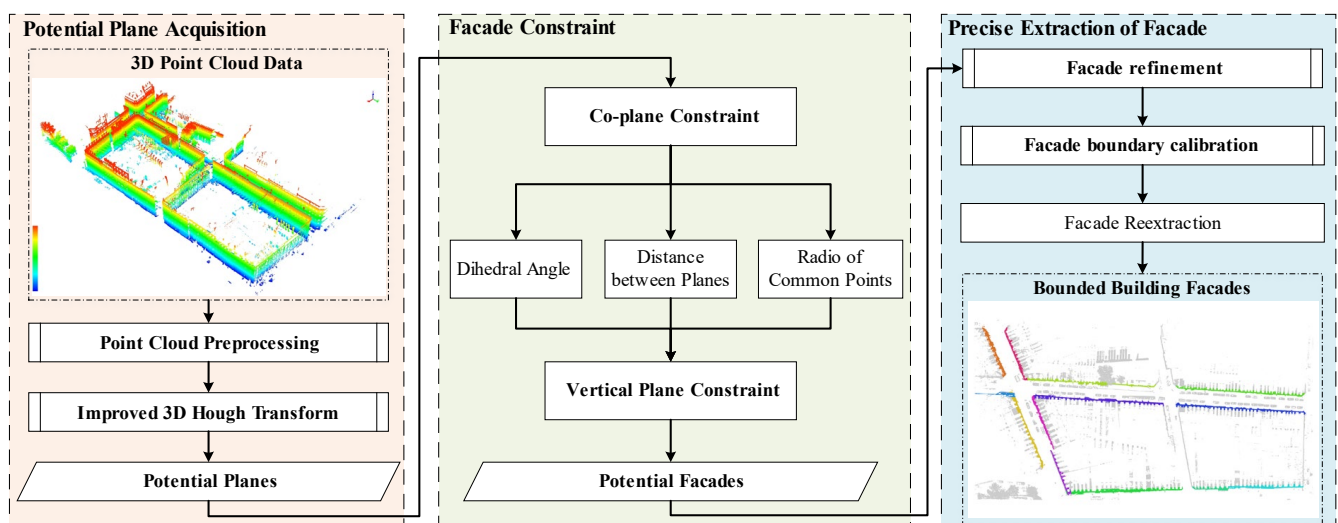


Figure 1. Processes of building facade extraction.

2.1.1. Improved 3D HT for Potential Plane Acquisition

Potential plane acquisition includes two steps: point cloud data preprocessing and plane equation extraction based on improved 3D HT. The purpose of preprocessing is to remove marked nonfacade point clouds and reduce the computation of subsequent algorithms. The improved 3D HT is primarily used for efficient and accurate potential plane acquisition.

(1) Point cloud data preprocessing.

Because point cloud data typically contain many ground points, and their density is typically high, these ground points must be removed first. In addition, the point cloud is panned to the origin of the coordinate system, and voxel downsampling is performed to reduce the computation volume. Eventually, statistical outlier removal is performed on the downsampled data to remove the point cloud noise.

(2) Improved 3D Hough transform.

This study aims to find a robust building facade extraction method to reduce data dependence. Thus, model-driven methods are more applicable than deep-learning-based point cloud segmentation methods, which are data-driven. For the plane detection of 3D PCD, Borrmann et al. proposed 3D HT based on the General Hough Transform (GHT) [25,26], which is a common model-driven method for point cloud plane detection. This method maps all planes that may pass through a point p_i into a surface in the Hough parameter space, with each point on the surface corresponding to a plane in the Cartesian coordinate system (Figure 2a). Multiple parametric surfaces form one or more intersections in the parametric space (Figure 2b). We thus count the number of planes that cross each intersection, and the plane corresponding to the highest cumulative number of intersections is the desired plane.

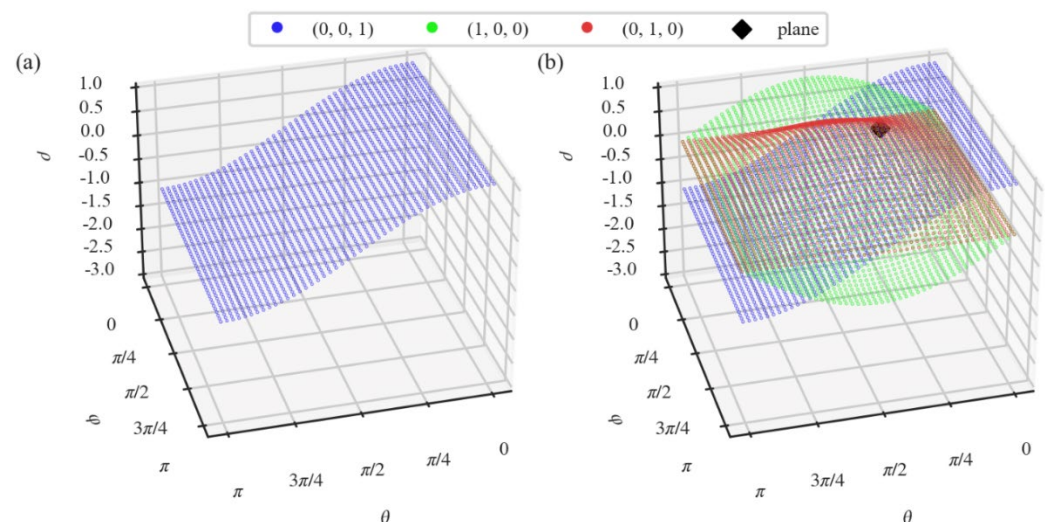


Figure 2. Transformation of three points from the Cartesian coordinate system (x, y, z) into Hough space (θ, ϕ, ρ) : (a) a surface corresponds to a point in the Cartesian coordinate system; and (b) three surfaces correspond to three points, and the black diamond intersection of the surfaces represents the plane spanning the three points.

For the 3D HT, the greatest challenge is choosing a step size. The discretization step sizes s_θ , s_ϕ , and s_ρ strongly affect plane extraction. Smaller discretization steps typically result in higher accuracies, but for every 1X increase in angular resolution (including s_θ and s_ϕ), the algorithm's computation and memory overhead increase by 4X, which is particularly important with large PCD. Thus, we propose using the shift vote strategy. In the discretization of plane parameters, ρ is discretized into the sets Q , and θ and ϕ are discretized into the sets $M\{0, s_\theta, 2s_\theta, \dots, 2\pi\}$ and $N\{0, s_\phi, 2s_\phi, \dots, 2\pi\}$ according to the

aforementioned strategy, respectively. Next, copies for each element offset by $s_\theta/2$ and $s_\varphi/2$ for $\mathbf{M}$ and $\mathbf{N}$ are created as:

$$\begin{cases} \mathbf{M}' = \{s_\theta/2, s_\theta + s_\theta/2, 2s_\theta + s_\theta/2 \dots, 2\pi + s_\theta/2\} \\ \mathbf{N}' = \{s_\varphi/2, s_\varphi + s_\varphi/2, 2s_\varphi + s_\varphi/2 \dots, 2\pi + s_\varphi/2\} \end{cases} \quad (1)$$

Thus, accumulators can be created as follows:

$$\begin{cases} \mathbf{A} = \mathbf{M} \times \mathbf{N} \times \mathbf{Q} = \{(\theta_j, \varphi_j, \rho_{ij}) | \theta_j \in \mathbf{M} \wedge \varphi_j \in \mathbf{N} \wedge \rho_{ij} \in \mathbf{Q}\} \\ \mathbf{A}' = \mathbf{M}' \times \mathbf{N}' \times \mathbf{Q} = \{(\theta'_j, \varphi'_j, \rho_{ij}) | \theta'_j \in \mathbf{M}' \wedge \varphi'_j \in \mathbf{N}' \wedge \rho_{ij} \in \mathbf{Q}\} \end{cases} \quad (2)$$

$\mathbf{A}$ and $\mathbf{A}'$ are voted on, and the candidate plane sets $\mathbf{S}(\theta, \varphi, \rho)$ and $\mathbf{S}'(\theta', \varphi', \rho)$ satisfying the conditions are obtained. Then, the concatenated set $\mathbf{S} \cup \mathbf{S}'$ is considered to be the final candidate plane set. Thus, the angular resolution is doubled, while the number of computations is only increased by $2\times$ and has the same memory overhead, achieving a balance of precision and efficiency. The optimal s_θ and s_φ are both 1° , which can achieve a good balance of precision and efficiency. The setting of s_ρ depends on the input data's range and the available memory. The recommended s_ρ range is 0.2~1 m. If s_ρ is larger than 1 m, the plane detection precision may be too low. Correspondingly, if s_ρ is smaller than 0.2 m, the marginal effect is marked, and the required memory increases without significant improvement in precision. Another major challenge is peak fuzziness, a prevalent issue with the Hough transform. Considering 2D HT as an example, a point in the Cartesian coordinate system corresponds to a curve in the Hough space. Theoretically, the parameter curves corresponding to points representing the same line should intersect at one point. Due to step settings, data noise, etc., these lines typically do not strictly intersect together (Figure 3; i.e., peak fuzziness), which causes difficulties for extraction, and this problem also exists in 3D HT. Therefore, we propose to perform 3D high-pass filtering on the accumulator of 3D HT to remove the low-frequency part of the accumulator and weaken the effect of peak fuzziness. The convolution kernel of 3D high-pass filtering is shown in Figure 4. The center cell of the convolution kernel is $1/2$, and other cells are determined according to the inverse distance weighted method, with a sum of 1. Finally, all potential planes are obtained by performing peak detection on the filtered accumulator.

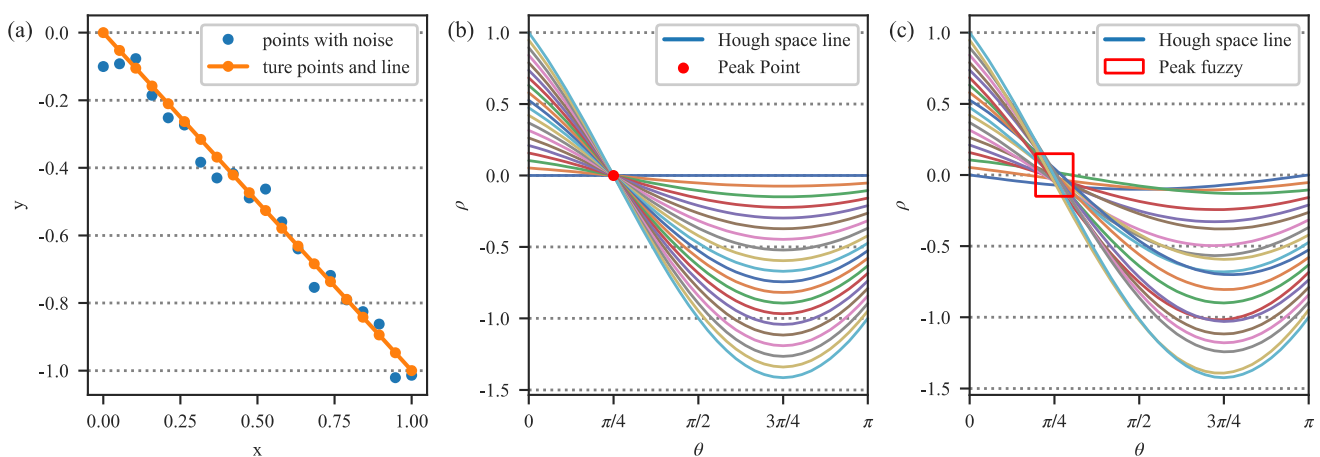


Figure 3. Schematic of peak fuzziness: (a) Cartesian coordinate system points; 2D points with noise (blue) bounce around true points (orange), and the orange line indicates the true line ($y = -x - 1$); (b) transforming the true points to the Hough parameter space, one line indicates one point in the Cartesian coordinate system, and the red dot indicates the intersection of all lines; (c) Hough parameter space with noise; transforming the points with noise to the Hough parameter space, one line indicates one point in the Cartesian coordinate system, and the red box shows the peak fuzziness.

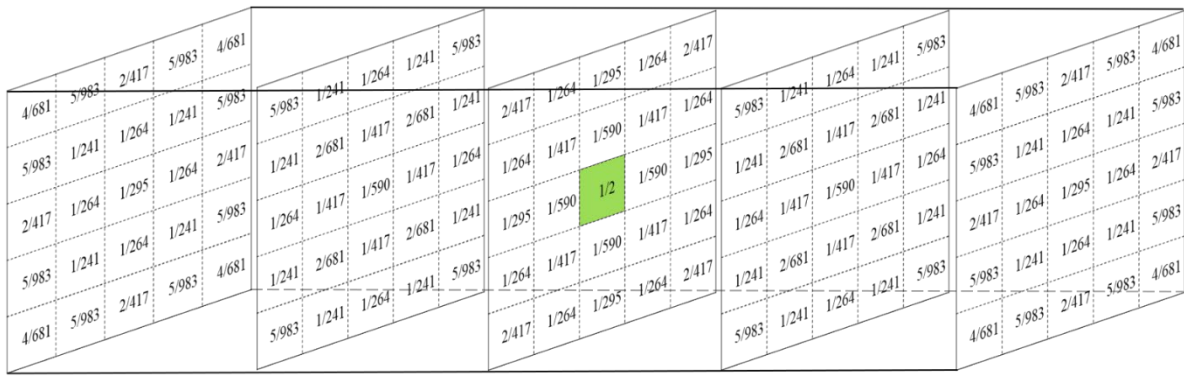


Figure 4. Schematic of the 3D high-pass filtering convolution kernel: the size of the convolution kernel is 5×5 ; its center pixel (green) is $1/2$; others are determined according to the distance from the pixel to the center based on the inverse distance weighted method; the sum of the pixels of the entire convolution kernel is 1.

2.1.2. Facade Constraints

Based on data quality, algorithm parameter settings, etc., the roughly extracted planes inevitably contain many pseudoplanes and nonbuilding planes. We introduce facade constraints to remove these planes and obtain the real building facades. This strategy includes coplane constraint and vertical plane constraint.

Coplane constraint. The purpose of coplane constraint is to eliminate the pseudoplanes caused by excessive point cloud density, inappropriate threshold setting, and peak fuzziness, which is primarily determined by three parameters: plane dihedral angle, plane distance [27], and common point ratio. If plane p_1 and p_2 satisfy:

$$((\arccos \frac{\mathbf{n}_1 \cdot \mathbf{n}_2}{|\mathbf{n}_1| |\mathbf{n}_2|} \leq \alpha_{th}) \wedge (\max(|\mathbf{r}_{12} \cdot \mathbf{n}_1|, |\mathbf{r}_{12} \cdot \mathbf{n}_2|) \leq \Delta d_{th})) \vee (ComProp(p_1, p_2) \leq cp_{th}) \quad (3)$$

They are regarded as coplanes and merged, where $\mathbf{r}_{12}$ is the distance vector of the origin's foot to lines r_1 and r_2 , which are perpendicular to planes p_1 and p_2 ; $\mathbf{n}_1$ and $\mathbf{n}_2$ are the normal vectors of planes p_1 and p_2 , respectively; $ComProp$ is the operator for estimating the proportion of common points between two planes based on the plane with fewer points; and α_{th} , Δd_{th} , and cp_{th} are the thresholds corresponding to plane dihedral angle, plane distance, and common point proportion, and are suggested to equal 5° , 1 m, 70%, respectively.

Vertical plane constraint. The improved 3D HT extracts all potential planes in the PCD, which contain both building facades and other planes. Because the building facade should be vertical, we eliminate other planes by constraining the vertical angle of each plane after the coplane constraint:

$$\arccos(\frac{\mathbf{m} \cdot \mathbf{n}}{|\mathbf{m}| |\mathbf{n}|}) \leq \alpha_{v,th} \quad (4)$$

where $\mathbf{m}$ and $\mathbf{n}$ are the normal vectors of the current plane and the vertical plane, respectively; and $\alpha_{v,th}$ is the threshold for the vertical plane constraint, where 75° can be used in most scenarios. A plane that does not meet the plane constraint will be discarded. After the coplane constraint and vertical plane constraint, the potential facades are obtained.

2.1.3. Precise Extraction of Facade

The result after facade constraint is still an infinitely extended plane in the mathematical sense. Figure 5a shows that two noncoplanar facades are regarded as one tilted plane, and Figure 5b shows that similar facades of different buildings are considered to be one plane. This situation does not meet the requirements of facade extraction and reduces the accuracy of the facade. Therefore, it is necessary to separate these point clouds in some

ways to obtain a clear range of the facade. In addition, nonplanar point clouds near the plane (e.g., trees, street lights, vehicles, and other feature point clouds) are easily mistaken for plane point clouds; thus, another role of facade precise extraction is to remove these nonplanar point clouds as much as possible. According to the goal of this study, facade precise extraction includes three parts: facade refinement, facade boundary calibration, and facade constraint. The facade constraint method is described in Section 2.1.2. This subsection introduces facade refinement and facade boundary delineation.

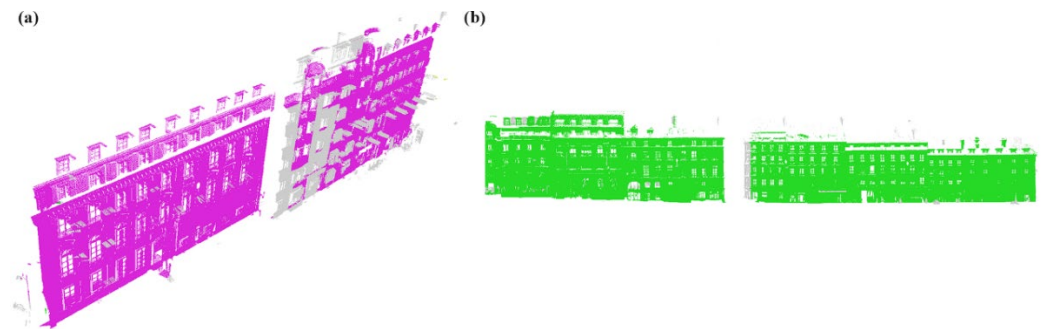


Figure 5. Different facades are identified as one: (a) two facades whose spatial locations are adjacent to each other are considered as one facade but are not coplanar; (b) similar facades of different buildings are incorrectly considered as one facade.

Facade refinement. We can enhance facade extraction accuracy by the improved 3D HT. However, its accuracy is still affected by parameter settings. The RANSAC method has the characteristics of low noise influence and no step size limitation, and can obtain a high accuracy facade from a large number of PCDs. Therefore, we propose a facade refinement strategy for further improving the quality of facade data resulting from the improved 3D HT. The overall process includes: (1) obtaining the plane equation of the potential facade corresponding to each point cloud cluster by RANSAC and acquiring the new potential facade; and (2) removing the coplanes and pseudoplanes by facade constraint to obtain the refined building facade. We focus on facade extraction, and the building facade is typically vertical (i.e., the plane of $C = 0$ in the planar general equation $Ax + By + Cz + D = 0$). Therefore, we set $C = 0$ when extracting the plane to improve the facade extraction accuracy.

Facade boundary calibration. Investigating the characteristics of point clouds, their density in each building's facade is found to be much higher than those in the interstices between facades, which agrees with the idea of the density-based clustering method. Therefore, the density-based clustering method is used for facade boundary calibration. The hierarchical density-based spatial clustering of applications with noise (HDBSCAN) method [28] is a commonly used cluster method and can cluster large-scale data robustly and efficiently. Therefore, the HDBSCAN method is used to cluster the point clouds after facade refinement. After clustering, the RANSAC method is implemented to extract facade equations and the corresponding facade point clouds from each point cloud cluster. The bounding boxes of the facade point clouds are considered to be facade boundaries. By performing the facade constraint described in Section 2.1.2 on the results of the facade boundary calibration, the bounded building facades and their corresponding point cloud data are obtained.

2.2. Building Facade Map Extraction

After obtaining the bounded building facades and their corresponding point cloud data, it is still difficult to extract facade maps from the disordered and unstructured point cloud data with varying qualities. Building facade maps can be divided into two parts: door and window boundaries and building boundaries. With the advancement of deep learning in recent years, the precision and speed of image object detection have markedly

improved [29–34], achieving better performance than traditional methods. The extraction of doors and windows is a form of object detection; thus, it is possible to use deep learning methods to extract window and door boundaries. Because building boundaries are typically large and complex in shape, it is difficult to extract them effectively using existing deep learning methods; thus, we chose to use digital image processing technology (e.g., image enhancement, filtering, edge detection, morphological processing, and connected domain analysis) to extract them according to the boundary features. By combining the extracted door and window boundaries with the building boundaries, the required building facade map is obtained. The proposed approach includes three steps: feature image generation, door and window detection, and building boundary extraction. The workflow of this process is shown in Figure 6.

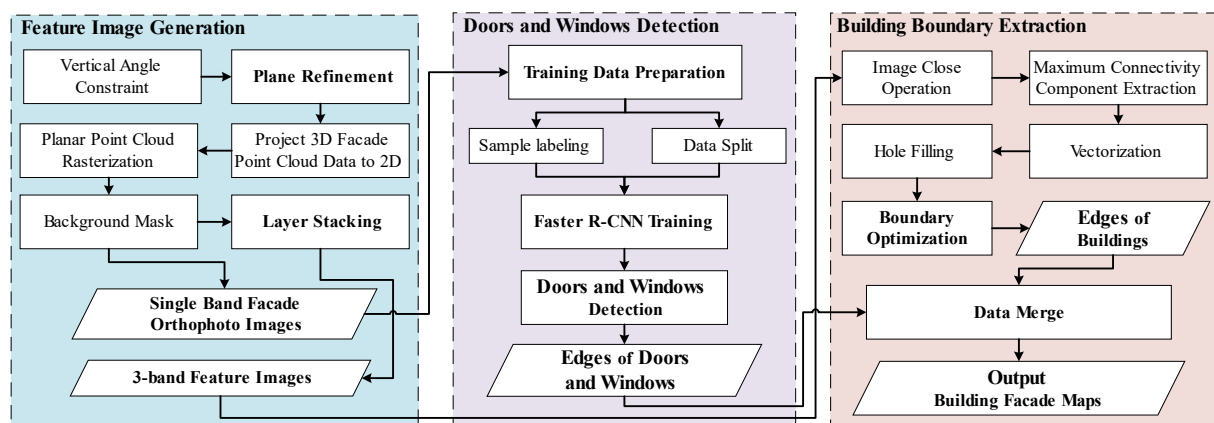


Figure 6. Processes of building facade map extraction.

Feature image generation. The building facade point cloud set $B(x, y, z)$ obtained from building facade extraction is converted into a point set $B'(x', y', z')$ with the corresponding facade as the reference coordinate system by the following equation:

$$\begin{cases} \alpha = \arctan(-A/B) \\ x' = \cos \alpha x + \sin \alpha y \\ y' = z \\ z' = \cos \alpha y - \sin \alpha x \end{cases} \quad (5)$$

where A and B are the plane general equation coefficients of the corresponding building facade. At this time, the height parameter z' of the point cloud relative to the plane is discarded, and the 2D plane point cloud $B''(x', y')$ is acquired by projecting the 3D point cloud onto the corresponding facade. A grid is created using the specified edge length, and the suggested length is less than 0.05 m to ensure edge extraction precision. Then, the 2D point cloud is divided by the grid. To rasterize the point cloud into a single-band 2D image, the number of point clouds within each grid is used as the pixel value. To enhance the image, histogram equalization, bandpass filtering after histogram equalization, and Prewitt edge detection are performed on the single-band 2D image. These processes can weaken the effects of uneven density among the point cloud and density differences between different point cloud datasets, improving the proposed method's robustness. A zero in the single-band feature image is masked before image enhancement to avoid its effect, particularly for histogram equalization and bandpass filtering. The resultant bands of the three operations are synthesized into a 3-band feature image, and the histogram equalization result is considered to be the single-band feature image. These two feature images are the final required feature images.

Doors and windows detection. In this study, we use a deep learning model to extract building doors, windows and their boundaries from 3-band feature images. Considering the characteristics of building door and window extraction, which requires high accuracy

without realtime processing, the Faster R-CNN model [33] is used for extraction because it has higher accuracy in the image object detection. The model architecture is shown in Figure 7. The most important feature of this model is the design of Region Proposal Networks. It generates the candidate regions using feature maps after the convolution operation. This can achieve higher detection speed and ensure higher accuracy compared to Selective Search, Edge Boxes, and other methods. Because there is only one target in the same plane and the same position of the facade, the detection results are nonmaximal suppressed, and only the results with the highest confidence are retained in the same position.

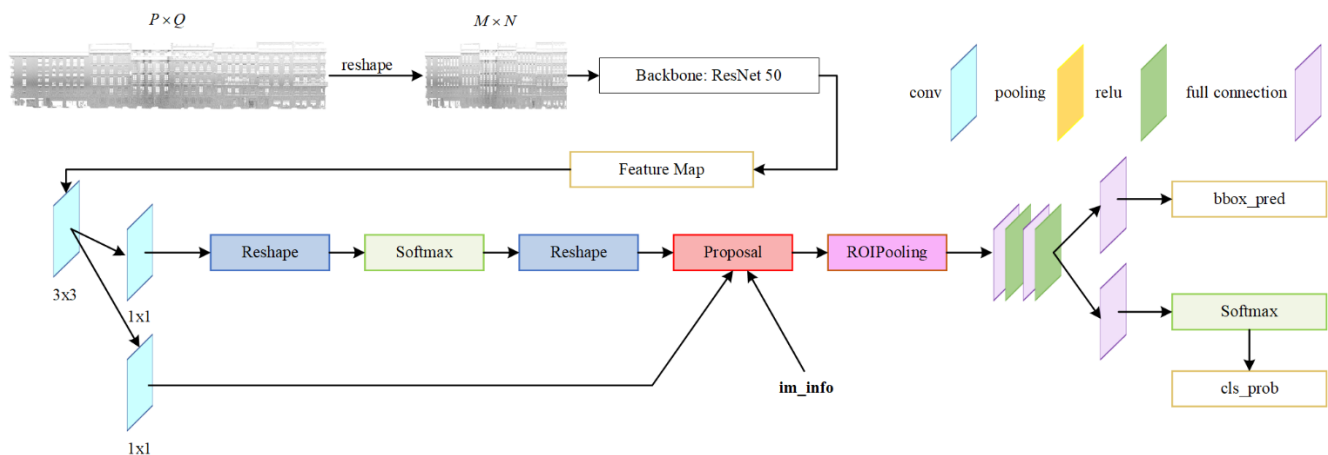


Figure 7. Architecture of Faster R-CNN. The cyan parallelogram, yellow parallelogram, green parallelogram, and purple parallelogram represent the convolutional layer, pooling layer, relu layer, and full connection layer, respectively. $P \times Q$ and $M \times N$ represent the height and width of the image. “cls_prob” represent the bounding box’s probability of various classes.

Building boundary extraction. Building boundaries are determined using digital image processing and optimization on the single-band feature images. The specific processes are: (1) performing a closing operation on the feature images to reduce noise and occlusion; (2) 8-adjacent connectivity domain detection; (3) vectorizing the maximum connectivity domain as the initial building boundary; (4) filling the void on the initial building boundary; and (5) boundary simplification and orthogonalization [35]. Finally, the optimized building boundaries are combined with the window and door boundaries to obtain the final building facade map.

3. Experiments and Results

In this section, the proposed method was evaluated on the IQmulus & TerraMobilita Contest dataset [36] and the Semantic3D.Net Benchmark dataset [37], which were obtained by mobile laser scanning (MLS) and terrestrial laser scanning (TLS), respectively. To verify the validity of the proposed method, the General Iterative RANSAC (GIR) and Vertical Constrained Iterative RANSAC (VCIR) methods were also used to extract the facade based on point cloud data. Section 3.1 introduces the results on the IQmulus & TerraMobilita Contest dataset, and Section 3.2 introduces the results on the Semantic3D dataset.

3.1. Results of the IQmulus & TerraMobilita Contest Dataset

The IQmulus & TerraMobilita Contest dataset [36] is 3D MLS data collected in Paris (France), consists of 300 million points, and covers approximately 10 km of streets within a square km of the 6th district of Paris. Most streets are covered in this square kilometer area; thus, the dataset is representative of this part of Paris. Due to the technology limitations of current laser scanning techniques, there are some problems with this dataset, such as uneven density, noise, occlusion, etc. In addition, there are stitching misalignments in some areas (Figure 8d). This dataset is often used for outdoor point cloud segmentation, from

which fast and accurate extraction of building facade maps remains a challenge. A subset of this dataset was selected as the experimental data, and the data range is shown in the red box in Figure 8.

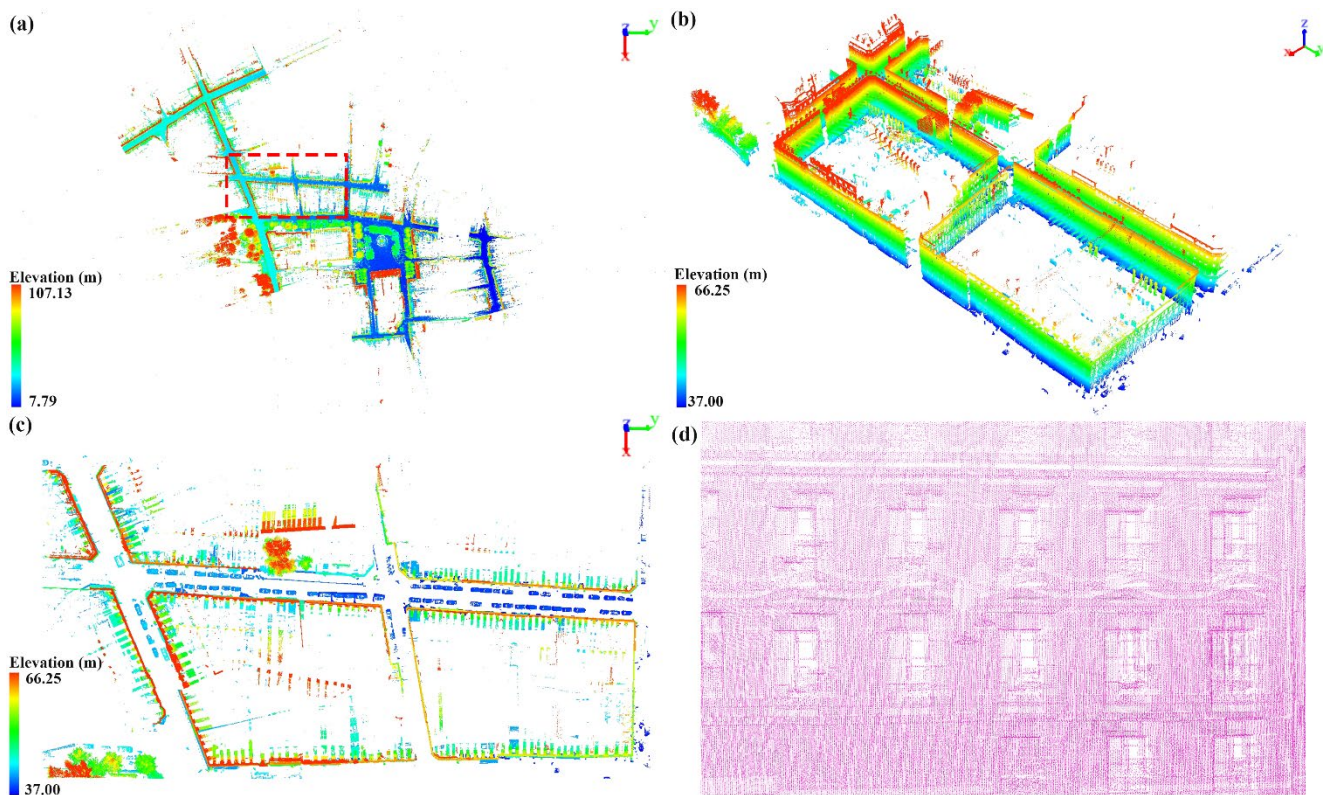


Figure 8. Raw experimental data for the IQmulus & Terra Mobilita Contest dataset: (a) the entire IQmulus & Terra Mobilita Contest dataset; the red dashed box shows the extent of the experimental data area; (b,c) the experimental data in 3D view and 2D view, respectively; and (d) sample areas with misalignment in the point cloud data.

Some preprocessing procedures were performed prior to facade extraction. First, the IQmulus & TerraMobilita Contest dataset is divided into nine data files and cannot be used directly. Thus, the nine files were merged into one file. Next, the point cloud data outside the subexperimental area were discarded, and the attributes other than coordinates (e.g., reflectivity and echo times) were removed to reduce the amount of data. Subsequently, these data were preprocessed as described in Section 2.1.1. The point cloud was voxel-downsampled with a 5 cm voxel side length, and the processed data consisted of 23 million points. Figure 8b,c show the processed point cloud data in 3D and 2D views, respectively.

Then, building facades were extracted from the preprocessed point cloud data by the method in Section 2.1, and the discretization step s_p of the improved 3D HT was set to 0.4 m. From the experimental data, thirteen planes were extracted using the proposed method. Seven and five planes were obtained using the GIR and VCIR methods, respectively. The extraction results of the three methods are shown in Figure 9. To distinguish these planes easily, we used Roman numerals I and II to number the planes extracted by the proposed and VCIR methods, respectively. Figure 9b shows that the GIR method cannot be applied to this scenario at all and fails to extract the correct building facades, which are all horizontal. Both the VCIR method and the proposed method can extract most facades from the experimental data. Therefore, we only compared the proposed method and VCIR method results.

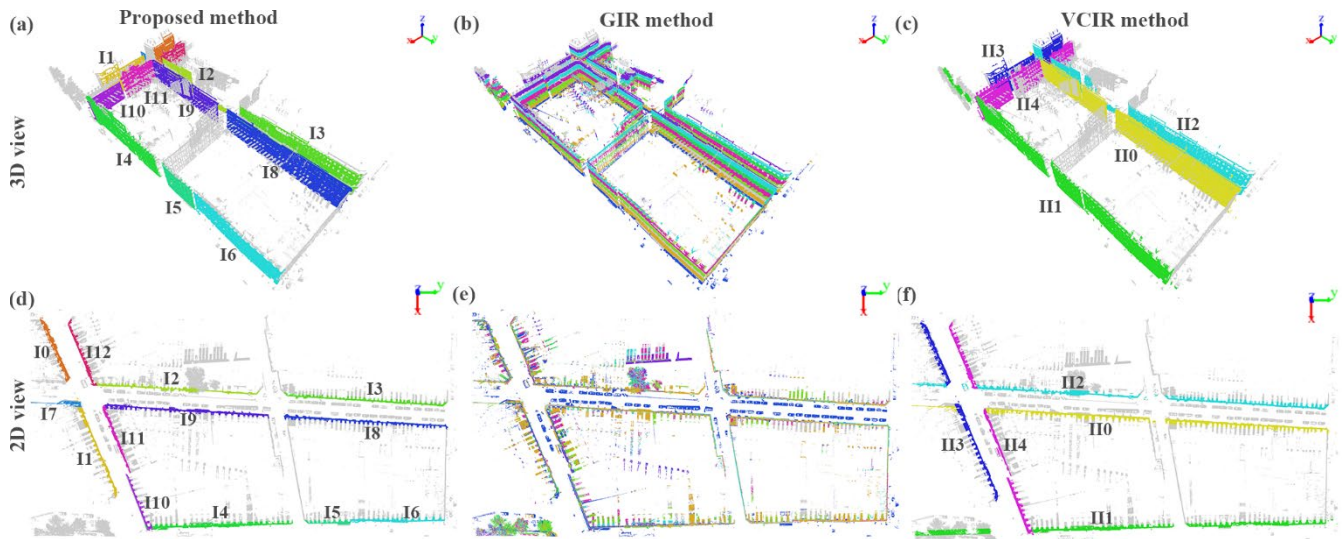


Figure 9. Facade extractions from the IQmulus & TerraMobilita Contest dataset: (a–c) the results in 3D view extracted by the proposed method, GIR method, and VCIR method, respectively; and (d–f) the results in 2D view extracted by the proposed method, GIR method, and VCIR method, respectively. Different colors were used for different facades and gray for nonfacade point clouds. Roman numeral I was used to number the facades extracted by the proposed method and Roman numeral II was used for the VCIR method.

To quantitatively evaluate the effects of these two methods, the mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE) were used to assess the facade extraction errors. For the single facade errors (SFE), the three accuracy indices were defined as:

$$\begin{cases} \text{MAE}_i = \left(\sum_{j=1}^n \frac{|A_i x_{ij} + B_i y_{ij} + C_i z_{ij} + D_i|}{\sqrt{A_i^2 + B_i^2 + C_i^2}} \right) / n \\ \text{MSE}_i = \left(\sum_{j=1}^n \frac{(A_i x_{ij} + B_i y_{ij} + C_i z_{ij} + D_i)^2}{A_i^2 + B_i^2 + C_i^2} \right) / n \\ \text{RMSE}_i = \sqrt{\text{MSE}_i} \end{cases} \quad (6)$$

where A , B , C , and D are the plane general equation coefficients of the building facade, and n is the number of points in the corresponding facade. Additionally, to evaluate the overall facade extraction errors (OFEE), the means of each facade error (MEFE) and overall facade error (OFE) were defined as:

$$\begin{cases} \text{MEFE} = \frac{\sum_{i=1}^m \text{SFE}_i}{m} \\ \text{OFE} = \frac{\sum_{i=1}^m (\text{SFE}_i m_i)}{\sum_{i=1}^m m_i} \end{cases} \quad (7)$$

where m is the number of the facades. The SFEs, MEFES, and OFEs of the two methods were calculated, and the results are shown in Table 1. Table 1 shows that the overall MAE and MSE of the proposed facade extraction method are 0.314 m and 0.194 m, respectively, which are only half of the overall errors of 0.500 m and 0.403 m of the VCIR method. Among the planes extracted by the proposed method, the minimum MSE is only 0.085 m, and the average is only 0.271 m, while the corresponding values of the VCIR method are 0.208 m and 0.439 m.

Table 1. Errors of facade extraction for the IQmulus & TerraMobilita Contest dataset.

Method	Number of Extract Facade	MAE	MSE	RMSE
Proposed method	I0	0.427	0.253	0.503
	I1	0.216	0.102	0.319
	I2	0.186	0.085	0.291
	I3	0.241	0.109	0.330
	I4	0.321	0.172	0.415
	I5	0.386	0.247	0.497
	I6	0.597	0.426	0.653
	I7	0.502	0.413	0.643
	I8	0.244	0.097	0.312
	I9	0.332	0.190	0.435
	I10	0.761	0.696	0.835
	I11	0.446	0.302	0.550
	I12	0.585	0.425	0.652
	MEFE	0.403	0.271	0.495
	OFE	0.314	0.194	0.440
VCIR method	II0	0.820	0.758	0.871
	II1	0.357	0.208	0.456
	II2	0.357	0.212	0.461
	II3	0.529	0.363	0.602
	II4	0.705	0.656	0.810
	MEFE	0.554	0.439	0.640
	OFE	0.500	0.403	0.634

To evaluate the effectiveness of the two methods in more detail, violin plots and bar plots were used to show the distribution of the MEFE for the two methods (Figure 10). As shown in Figure 10a, the median and quartiles of the three errors of the proposed method are smaller than those of the VCIR method. The probability density distribution of the proposed method also looks like a fusiform, which is narrow at the top and wide at the bottom, and the error is low overall. In contrast, the probability density distribution of the VCIR method is gourd-shaped, and the error distribution is relatively uniform. The standard deviation and median value of the MAE extracted by the proposed method are 0.17 and 0.39, respectively, which are markedly lower than the corresponding 0.21 and 0.53 extracted by the VCIR method. As shown in Figure 10b, the average errors of the three errors of the proposed method are smaller than those of the VCIR method, and the error distribution is more concentrated and stable. Thus, the proposed method outperforms the VCIR method in terms of facade extraction accuracy for the IQmulus & TerraMobilita Contest dataset.

Next, building facade map extraction was performed based on the building facades obtained by the proposed method. First, the feature images were generated as described in Section 2.2 and the edge length for feature image generation was set to 0.02 m. The windows and doors in the generated 3-band feature images were labeled using the ArcGIS Pro platform, and a total of 948 windows and 53 doors were obtained. Then, the feature images were sliced using these labeled windows and doors with a 256×256 slicing size and 50% overlap. Because windows and doors have typical horizontal and vertical characteristics, rotational transformations were not used for the sliced results. A total of 2835 sets of samples were generated after slicing, and these samples were separated into a training set and a validation set at a 7:3 ratio. Due to the small number of samples included in each facade, the testing set was not generated, but the entire sample of each facade was used for the final model accuracy evaluation. The next step established and trained the Faster R-CNN model. The ResNet50 pretraining model was used as a backbone, the training batch size was 4, and the number of training epochs was 200. For the learning rate determination, Smith proposed the cyclical learning rates (CLR) method [38], achieving the best training status of model parameters without adding additional computation. This method was used to determine the best learning rate of the model, and the optimal learning

rate was 1.096×10^{-4} . Finally, the door and window boundaries were extracted from all building facade feature images using the trained model. The final building facade maps were obtained by combining the final door and window boundaries with the building boundaries obtained by the digital image processing method.

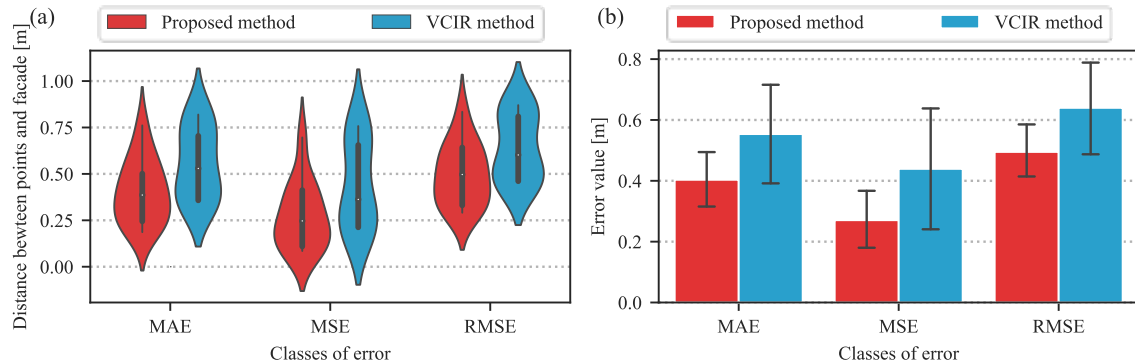


Figure 10. Violin plots and bar plot for the errors of facade extraction for the IQmulus & TerraMobilita Contest dataset: (a) violin plot of the distances between points and facade, where the shape of the violin displays the probability density distribution of the data; the black bar depicts the interquartile range, the 95% confidence interval is shown by the inner line branching from it, and the median is shown by a white dot; and (b) bar plot of the total facades error, where the vertical bars show the mean of the data and the error line shows the 95% confidence interval.

Considering the three typical facades I3, I8, and I9 as examples, the facade map extraction results are shown in Figure 11. As shown in the figure, the vast majority of doors and windows were detected successfully, and the extracted door and window boundaries are primarily distributed horizontally and vertically, agreeing with the real door and window boundaries. The accuracy of building boundary extraction is marginally lower, while the overall trend matches the real trend, which means that the extraction effect of the proposed method meets the expectation.

Additionally, the results of window extraction are evaluated quantitatively. Precision, recall, F1 score, accuracy, confusion matrix, average precision (AP), intersection over union (IoU), and other indices are often used to evaluate the accuracy of results in the field of object detection. Because the window extraction in this paper is for single-category object detection, the confusion matrix and other indicators for multicategory object detection are not applicable; therefore, the remaining four indicators are used for the facade. The extraction accuracy for each facade is shown in Table 2. The overall accuracy, recall, and F1 score of window rough extraction with min IoU set at 50% reached 0.982, 0.977, and 0.979, respectively, which means that the vast majority of windows can achieve rough extraction correctly. For window precise extraction with min IoU set at 85%, the overall accuracy, recall, and F1 score reached 0.887, 0.882, and 0.884, respectively. The minimum F1 score and AP were 0.774 and 0.621, respectively, and the corresponding averages were 0.990 and 0.827, respectively, which means that most windows can obtain accurate edges.

To describe the extraction accuracy in more detail, box plots were drawn using each facade map extraction accuracy index (Figure 12). Figure 12a shows that, at a minimum IoU of 50%, the medians of the four accuracy indices all exceed 0.97, the lower quartiles all exceed 0.96, and the precision averages are close to 1.00. Figure 12b shows that, at a minimum IoU of 85%, the median of accuracy, recall, and the F1 score are all over 0.90, and the lower quartiles are all over 0.85. Thus, the proposed method can obtain a good result with the MLS dataset, producing a rough extraction of nearly all parts of the windows and accurate extraction of most of the windows.



Figure 11. Facade map extraction with the IQmulus & TerraMobilita Contest dataset: the number in the upper left corner corresponds to the facade number in Figure 9; the red, blue, and green lines in the figure represent the window, door, and building boundaries, respectively; and the background image shows the single-band feature image, and the darker the image pixel color, the greater the number of points contained in the corresponding planar grid.

Table 2. Accuracy of window extraction for the IQmulus & TerraMobilita Contest dataset.

ID of the Facade	MinIoU: 50%				MinIoU: 85%			
	Precision	Recall	F1 Score	AP	Precision	Recall	F1 Score	AP
I0	1.000	1.000	1.000	1.000	0.933	0.933	0.933	0.871
I1	1.000	1.000	1.000	1.000	0.947	0.947	0.947	0.898
I2	1.000	0.974	0.987	0.974	0.921	0.897	0.909	0.827
I3	0.986	0.959	0.972	0.952	0.845	0.822	0.833	0.719
I4	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
I5	0.986	0.986	0.986	0.986	0.890	0.890	0.890	0.827
I6	0.912	0.981	0.945	0.945	0.842	0.906	0.873	0.805
I7	1.000	0.976	0.988	0.976	0.901	0.880	0.890	0.793
I8	0.964	1.000	0.981	0.964	0.891	0.925	0.907	0.839
I9	1.000	1.000	1.000	1.000	0.949	0.949	0.949	0.900
I10	1.000	0.938	0.968	0.938	0.959	0.899	0.928	0.862
I11	0.947	0.992	0.969	0.969	0.848	0.889	0.868	0.784
I12	1.000	0.987	0.994	0.987	0.779	0.769	0.774	0.621
Mean value of each facade	0.984	0.984	0.984	0.976	0.901	0.900	0.900	0.827
All facade	0.982	0.977	0.979	-	0.887	0.882	0.884	-

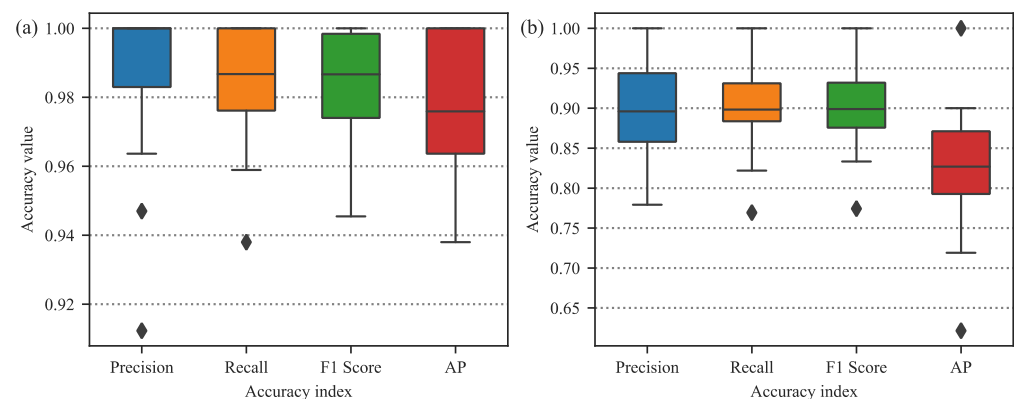


Figure 12. Box plot of the accuracy of window extraction for the IQmulus & TerraMobilita Contest dataset: (a) the accuracy with a min IoU of 0.5; and (b) the accuracy with a min IoU of 0.85. The different colored boxes represent different precision indicators. The upper and lower quartiles of the data are shown by the box's upper and lower boundaries, respectively, and the median is shown by the inner horizontal line. The whiskers extending from the ends of the boxes are used to represent variables other than the upper and lower quartiles, and outliers are represented by black dots.

3.2. Results of the Semantic3D.Net Benchmark Dataset

The Semantic3D.Net Benchmark dataset [37] is a 3D TLS point cloud dataset that was scanned statically with modern equipment and contains fine details. This dataset contains over four billion points and covers a range of diverse urban scenes, such as churches, streets, railroad tracks, squares, villages, soccer fields, castles, etc. In this paper, the “domfountain” and “marketsquarefeldkirch” subsets of this dataset were used to evaluate the proposed method for TLS point cloud data and were primarily collected in the cathedral and market square. All but the coordinates have been deleted to reduce the amount of data. Then, these data were preprocessed as described in Section 2.1.1. The point cloud was voxel-downsampled with a 5 cm voxel side length, and the processed data of these two scenes contained 76 million points and 38 million points, respectively. The processed point cloud data are shown in Figure 13.

Then, building facades were extracted from the processed point cloud data by the method in Section 2.1, and the discretization step s_p of the improved 3D HT was set to 0.4 m. The extraction results of the three methods are shown in Figure 14. For the “domfountain” scene, the proposed method and VCIR method extracted nine and four planes, respectively. For the “marketsquarefeldkirch” scene, nine and four planes were extracted by the proposed method and VCIR method, respectively. To distinguish these planes easily, we use the capital English letters A, B, C, and D to number the planes extracted by the proposed and VCIR method for the “domfountain” scene and “marketsquarefeldkirch” scene, respectively. Figure 14 shows that the GIR method cannot be applied to this scenario at all, and the VCIR method and the proposed method can extract most facades from the experimental data. The results are thus similar to those with the IQmulus & TerraMobilita Contest dataset. Therefore, we only compared the proposed method and VCIR method in the following.

Additionally, the MAE, MSE, and RMSE were used to evaluate the facade extraction errors quantitatively, and the results are shown in Table 3. In the “domfountain” scene, the overall MAE and MSE of the proposed facade extraction method are 0.335 m and 0.222 m, respectively; the overall MAE and MSE of the proposed facade extraction method for the “marketplacefeldkirch” scene are 0.296 m and 0.198 m, respectively. The two scenes' total errors are only half of the corresponding overall errors extracted by the VCIR method. To evaluate the effectiveness of the two methods for the TLS point cloud data in more detail, violin plots and bar plots were used to show the distribution of the MEFE for the two methods (Figure 15). As shown in Figure 15a,c, the median and quartiles of the three errors of the proposed method are smaller than those of the VCIR method, and the errors of the proposed method mostly are typically small, while the errors of the VCIR method are

evenly distributed. The standard deviation and median of the MAE for the “marketplace-feldkirch” scene extracted by the proposed method are 0.16 and 0.37, respectively, which are markedly lower than the corresponding values of 0.25 and 0.64 extracted by the VCIR method. As shown in Figure 15b,d, the average errors of the three errors of the proposed method are smaller than those of the VCIR method, and the error distribution is more concentrated and stable. Thus, the proposed method outperforms the VCIR method in terms of facade extraction accuracy for the Semantic3D.Net Benchmark dataset.

Next, feature image generation and door and window sample labeling were performed in the same way as with the IQmulus & TerraMobilita Contest dataset. A total of 482 windows and 98 doors were obtained, and a total of 898 sets of samples were generated for the two scenes. Additionally, these samples are separated into a training set and a validation set at a 7:3 ratio. The testing set was not generated, while the entire sample of each facade was used for the final model accuracy evaluation. The Faster R-CNN model was created and trained in the same way as for the IQmulus & TerraMobilita Contest dataset. The door and window boundaries were then extracted from all building facade feature images using the trained model. The final building facade maps were obtained by combining the final door and window boundaries with the building boundaries obtained by the digital image processing method. The facade map extraction results are shown in Figure 16. Figure 16a,e show the facade maps of facades A1 and C2, respectively. The vast majority of windows have been detected successfully, and the extracted window boundaries are primarily distributed horizontally and vertically, which are in good agreement with the real boundaries.

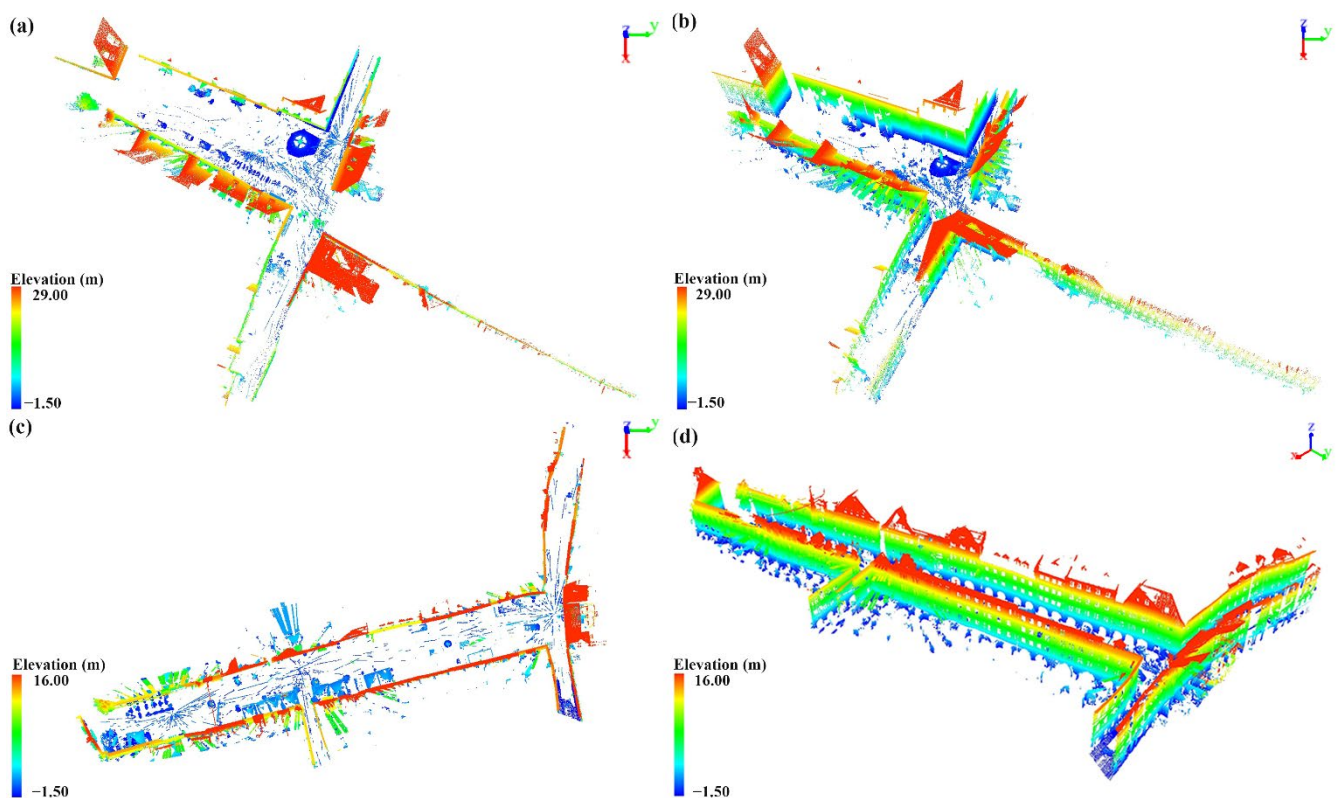


Figure 13. Raw experimental data for the Semantic3D.Net Benchmark dataset: (a,b) the processed point cloud data of the “domountain” scene in 2D view and 3D view, respectively; and (c,d) the processed point cloud data of the “marketplacefeldkirch” scene in 2D view and 3D view, respectively.

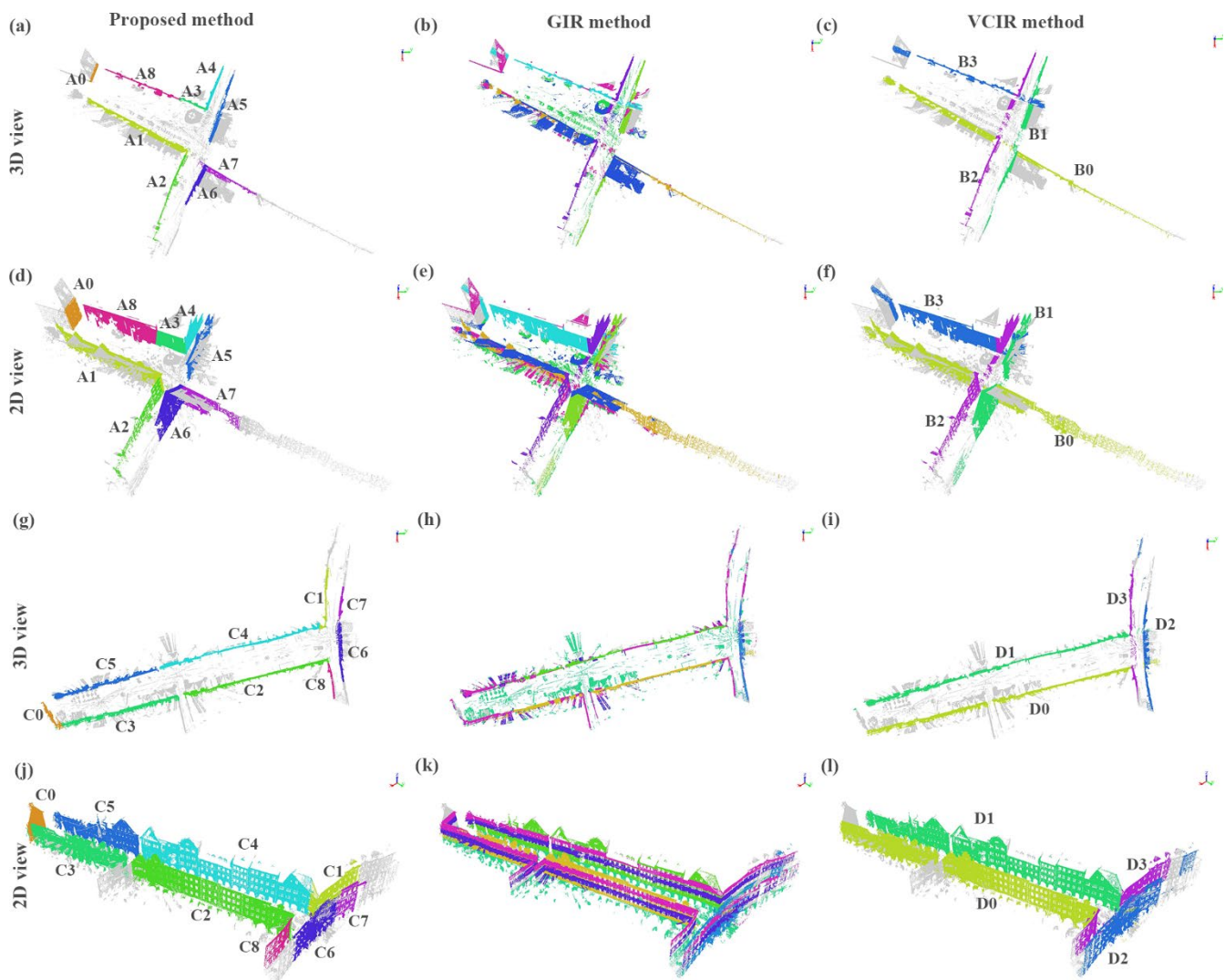


Figure 14. Facade extraction with the Semantic3D.Net Benchmark dataset: (a–c) results of the “domfountain” scene in 2D view extracted by the proposed method, GIR method, and VCIR method, respectively; (d–f) results of the “domfountain” scene in 3D view extracted by the proposed method, GIR method, and VCIR method, respectively; (g–i) results of the “marketsquarefeldkirch” scene in 2D view extracted by the proposed method, GIR method, and VCIR method, respectively; and (j–l) results of the “marketsquarefeldkirch” scene in 3D view extracted by the proposed method, GIR method, and VCIR method, respectively. Different colors were used for different facades and gray was used for nonfacade point clouds. Capital English letters were used to label these facades.

Additionally, precision, recall, the F1 score, AP, and IoU were used to quantitatively evaluate the results of window extraction. The window extraction’s accuracy for each facade is shown in Table 4. The overall accuracy, recall, and the F1 score of window rough extraction with min IoU set to 50% for the “domfountain” scene reach 0.936, 0.970, and 0.953, respectively. The corresponding accuracy indices of window rough extraction for the “marketplacefeldkirch” scene reach 0.981, 0.984, and 0.983, respectively. These results indicate that the vast majority of windows can be roughly extracted correctly. For window precise extraction with min IoU set to 85%, the overall accuracy, recall, and F1 score for the “domfountain” scene reach 0.884, 0.916, and 0.900, respectively. The corresponding accuracy indices of window precise extraction for the “marketplacefeldkirch” scene reach 0.962, 0.965, and 0.964, respectively. These results indicate that most windows are given accurate edges.

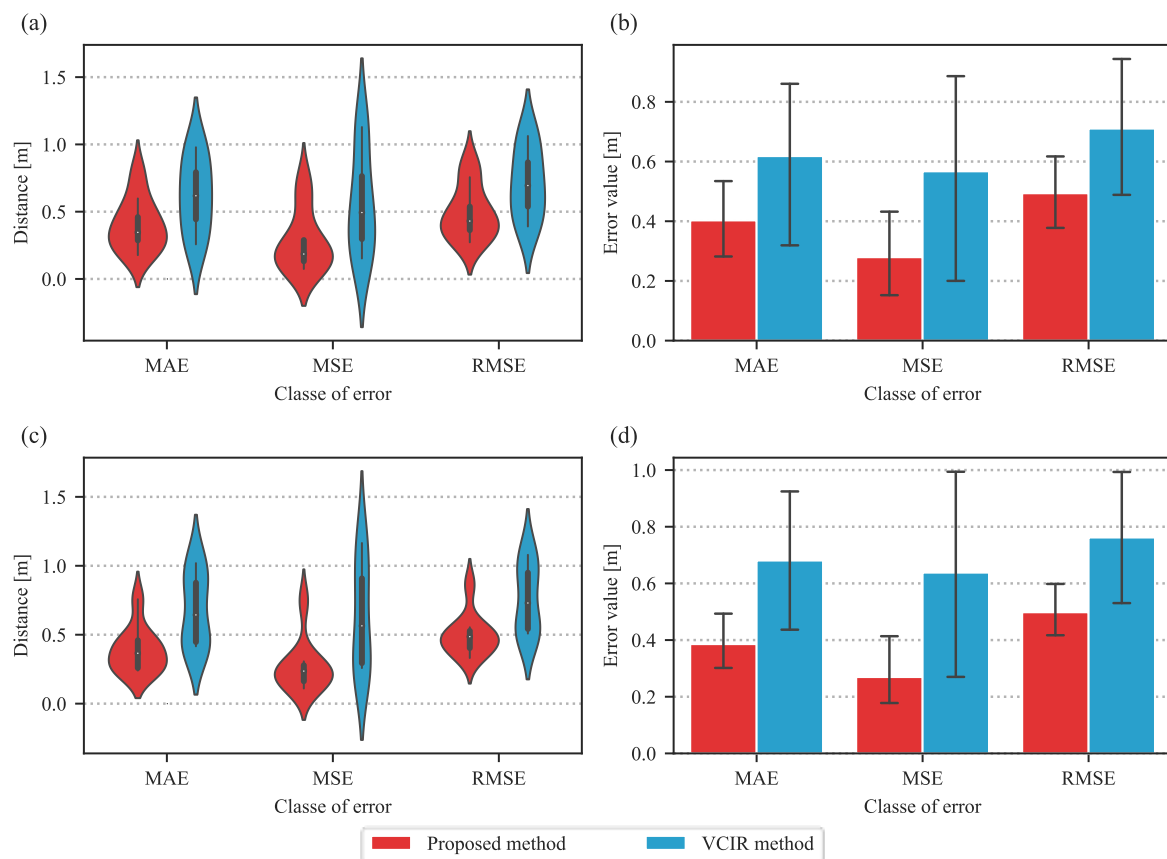


Figure 15. Violin plots and bar plot for the errors of facade extraction for Semantic3D.Net Benchmark dataset: (a,c) violin plots for distances between points and facades for the “domfountain” and “marketplacefeldkirch” scenes, respectively, where the shape of the violin describes the data’s probability density distribution; the black bar describes the interquartile range; the 95% confidence interval is shown by the inner line branching from it; and the median is shown by a white dot; and (b,d) bar plots for total errors of the facades for the “domfountain” and “marketplacefeldkirch” scenes, respectively, where the vertical bars show the mean of the data, and the error line shows the 95% confidence interval.

Table 3. Errors of facade extraction for the Semantic3D.Net Benchmark dataset.

Scene Name	Method	Number of Extracted Facade	MAE	MSE	RMSE
domfountain	Proposed method	A0	0.204	0.074	0.272
		A1	0.287	0.151	0.389
		A2	0.598	0.572	0.756
		A3	0.177	0.094	0.306
		A4	0.346	0.185	0.430
		A5	0.460	0.281	0.531
		A6	0.299	0.131	0.363
		A7	0.460	0.289	0.538
		A8	0.793	0.736	0.858
		MEFE	0.403	0.279	0.494
		OFE	0.335	0.222	0.471
	VCIR method	B0	0.979	1.129	1.063
		B1	0.257	0.152	0.390
		B2	0.505	0.344	0.587
		B3	0.731	0.643	0.802
		MEFE	0.618	0.567	0.710
		OFE	0.494	0.418	0.647

Table 3. Cont.

Scene Name	Method	Number of Extracted Facade	MAE	MSE	RMSE
marketplacefeldkirch	Proposed method	C0	0.462	0.307	0.554
		C1	0.757	0.746	0.864
		C2	0.243	0.144	0.379
		C3	0.417	0.246	0.496
		C4	0.366	0.236	0.486
		C5	0.240	0.162	0.403
		C6	0.267	0.195	0.442
		C7	0.259	0.110	0.331
		C8	0.460	0.279	0.528
		MEFE	0.386	0.269	0.498
		OFE	0.296	0.198	0.445
	VCIR method	D0	0.457	0.306	0.553
		D1	0.416	0.258	0.508
		D2	1.019	1.165	1.079
		D3	0.830	0.823	0.907
		MEFE	0.681	0.638	0.762
		OFE	0.464	0.385	0.621

Table 4. Accuracy of window extraction for Semantic3D.Net Benchmark dataset.

Scene Name	ID of the Facade	Min IoU: 50%				Min IoU: 85%			
		Precision	Recall	F1 Score	AP	Precision	Recall	F1 Score	AP
domfountain	A0	1.000	1.000	1.000	1.000	0.909	0.909	0.909	0.826
	A1	0.981	1.000	0.991	0.981	0.981	1.000	0.991	0.981
	A2	0.958	0.958	0.958	0.918	0.875	0.875	0.875	0.766
	A3	1.000	0.667	0.800	0.667	0.000	0.000	0.000	0.000
	A4	0.167	0.500	0.250	0.100	0.000	0.000	0.000	0.000
	A5	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	A6	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	A7	1.000	1.000	1.000	1.000	0.952	0.952	0.952	0.907
	A8	0.333	0.500	0.400	0.200	0.000	0.000	0.000	0.000
	Mean value of each facade	0.888	0.891	0.875	0.833	0.715	0.717	0.716	0.685
marketplacefeldkirch	All facade	0.936	0.970	0.953	-	0.884	0.916	0.900	-
	C0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	C1	0.810	1.000	0.895	0.895	0.810	1.000	0.895	0.895
	C2	1.000	0.986	0.993	0.986	0.986	0.972	0.979	0.958
	C3	1.000	1.000	1.000	1.000	0.962	0.962	0.962	0.925
	C4	0.973	0.973	0.973	0.947	0.960	0.960	0.960	0.934
	C5	1.000	1.000	1.000	1.000	0.972	0.972	0.972	0.972
	C6	1.000	0.944	0.971	0.944	1.000	0.944	0.971	0.944
	C7	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	C8	1.000	1.000	1.000	1.000	0.917	0.917	0.917	0.840
	Mean value of each facade	0.976	0.989	0.981	0.975	0.956	0.970	0.962	0.941
	All facade	0.981	0.984	0.983	-	0.962	0.965	0.964	-

To describe extraction accuracy, violin plots were drawn using each facade map extraction accuracy index (Figure 17). Considering Table 4, Figure 17a,b, the distribution of the window extraction accuracy for the “domfountain” scene is not uniform. The accuracy of the vast majority of facades is high, while the accuracy of facades A3, A4 and A8 is near zero. These results are primarily due to the difference between these facades’ window and door shapes and other facades’ window and door shapes. For the window extraction accuracy for the “marketplacefeldkirch” scene, at a minimum IoU of 50%, the lower quartiles all exceed 0.95, and the medians of the four accuracy indices all reach 1.0 (Figure 17c). At a minimum IoU of 85%, the lower quartiles all exceed 0.92, and the medians of the four accuracy indices all exceed 0.94 (Figure 17d). The facades with relatively low accuracy are facades C3 and C5, which is primarily due to the misalignment of the point cloud data (Figure 16f,g). Thus, the proposed method can obtain good results with the

TLS point cloud data, producing rough extractions of nearly all parts of the windows and accurate extractions of most windows.

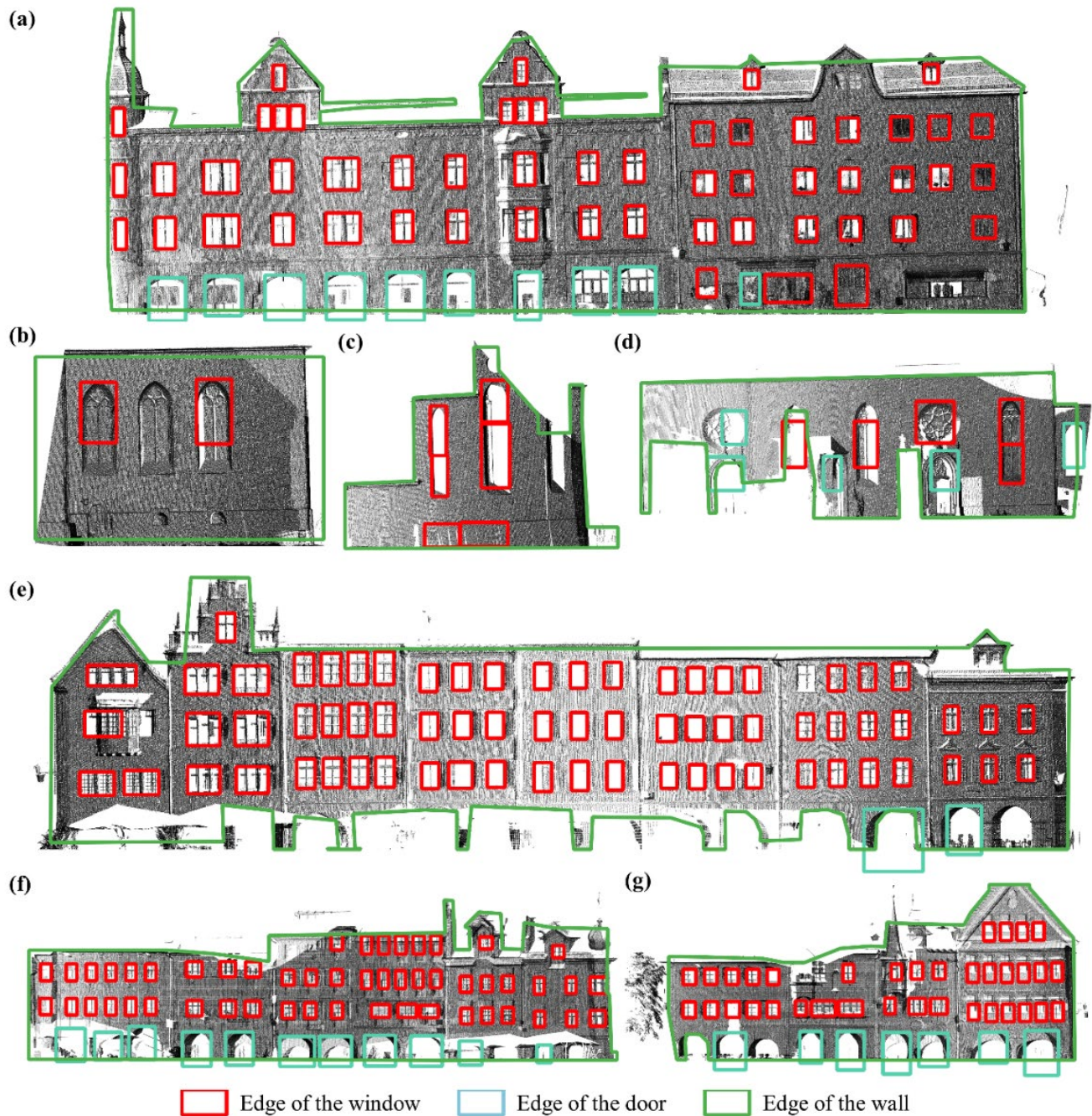


Figure 16. Facade map extraction with the Semantic3D.Net Benchmark dataset: (a–d) the facade map extraction results for the “domfountain” scene of facades A1, A3, A4, and A8, respectively; and (e–g) the facade map extraction results for the “marketplacefeldkirch” scene of facades C2, C3, and C5, respectively. The red, blue, and green lines in the figure represent the window, door, and building boundaries, respectively; the background image shows the single-band feature image; and the darker the image pixel color is, the greater the number of points contained in the corresponding planar grid.

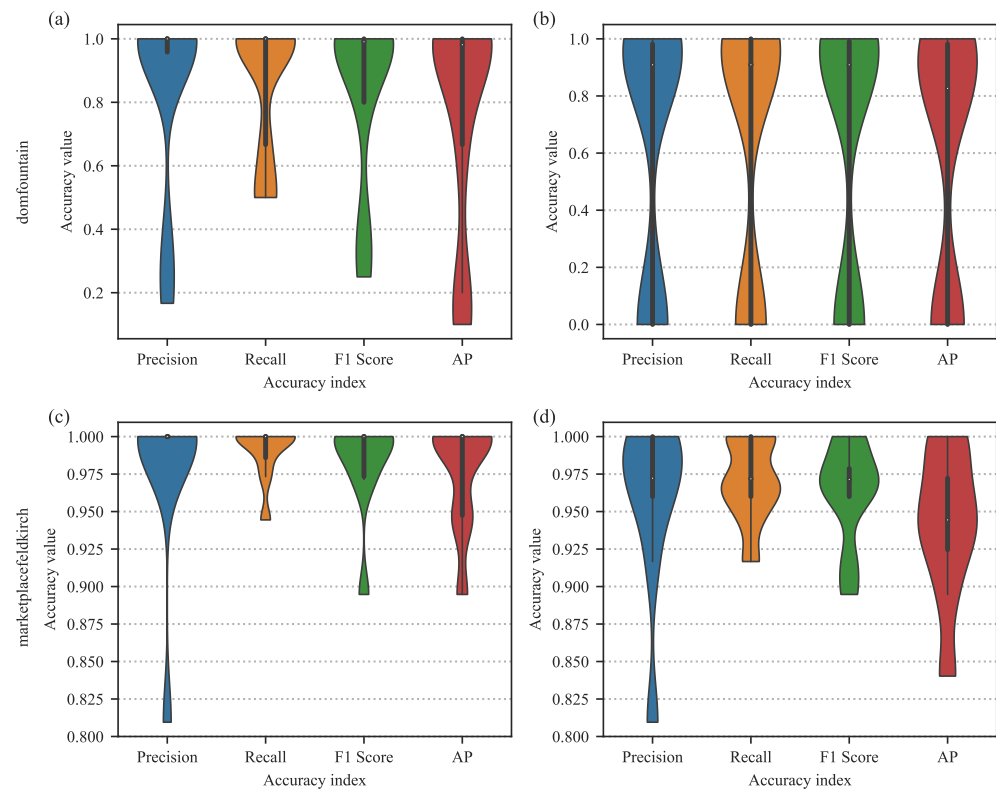


Figure 17. Box plot of the accuracy of window extraction with the Semantic3D.Net Benchmark dataset: (a,c) the accuracy with a minimum IoU of 0.5 of the “domfountain” and “marketplacefeldkirch” scenes, respectively; and (b,d) the accuracy with a minimum IoU of 0.85 of the “domfountain” and “marketplacefeldkirch” scenes, respectively. The different colored plots represent different precision indicators. The shape of the violin displays the data’s probability density distribution; the black bar depicts the interquartile range; the 95% confidence interval is shown by the inner line branching from it; and the median is shown as a white dot.

4. Discussion

Building facade map extraction is an important research topic in point cloud information extraction, and many studies have proposed different extraction methods from different perspectives. For example, slicing methods can detect windows and doors of any shape [8,11]. Maas and Vosselman proposed two algorithms to extract building models based on triangular meshes and plane intersections [14]. All of these methods are elegant and can achieve good results on good datasets. However, due to the complexity of the real world and various problems in data acquisition and processing, the collected point cloud data inevitably contain many problems, such as occlusion and misalignment. These problems make it difficult to achieve good results with traditional methods. Deep-learning-based methods have powerful information extraction capabilities, and many deep-learning-based point cloud segmentation and information extraction models have been proposed [19–21,39]. However, there are still many challenges to obtain building facade maps directly from the 3D PCD using deep-learning-based methods. First, the 3D PCD is unstructured and large in volume, which increases the difficulty and complexity of information extraction. In addition, deep-learning-based methods of point cloud processing are highly data-dependent and difficult to adapt to data collected by different approaches for different cities, different scenes, and different densities and occlusions.

Considering these data problems, this paper proposes a novel building facade map extraction method to improve the quality of facade map extraction from point cloud data with poor quality. For example, the proposed method combines traditional model-driven methods (the 3D HT and RANSAC methods) and the data-driven deep-learning-based

method. Model-driven methods have been used to extract the facades from the 3D PCD and generate the feature images. Thus, the unstructured 3D PCD has been transformed into structured 2D feature images. These processes decreased the data dimension and complexity and reduced the variability between different datasets. Then, a deep-learning-based object detection methods was used to obtain the window and door boundaries based on the 2D feature images. This method can learn different cases regarding data problems such as occlusion and misalignment, improving robustness. In addition, for facade extraction, many traditional methods, including the VCIR method described in this paper, treat facades as mathematical planes, which extend infinitely in space; thus, it is inevitable to identify similar adjacent facades as the same facade. Figure 9c,d and Table 5 show that one plane extracted by the VCIR method typically corresponds to multiple facades that are extracted by the proposed method. Thus, considering the bounded facades, the proposed method performs the facade boundary calibration based on the HDBSCAN algorithm after the initial facade extraction. Because the point cloud density inside each building facade is typically high, while the density inside the gap between facades is low, the HDBSCAN method which is based on density clustering can effectively distinguish adjacent facades. Thus, this method improves the accuracy of facade extraction and can remove the point clouds of nonfacades. Figure 18 shows the advantages of the proposed method in facade discrimination through partial enlargement. Different colors represent different facades, and gray point clouds represent nonfacade point clouds. Considering Table 5 and Figure 18c,d, the proposed method can correctly distinguish the three facades I10, I11, and I12, while the VCIR method incorrectly considers these three facades to be the same facade (facade I14). Figure 18a,b show that the maximum distances from the point cloud on facades I10 and I11 extracted by the proposed method to the corresponding facade are 0.93 m and 0.37 m, respectively. The error of the proposed facade extraction method is near half that of the VCIR method. In addition, the VCIR method regards point clouds of other objects adjacent to the facade as facade point clouds. For example, tree point clouds are calibrated as facade I11's point clouds (i.e., the red dotted box in the lower-left corner shown in Figure 18c), while the proposed method can correctly distinguish them (Figure 18b).

As mentioned above, occlusion and misalignment are common quality problems in point cloud data that can strongly impact facade map extraction. To reduce their impact on the facade map extraction results, the Faster R-CNN model is used to implement window and door boundary extraction. This model has a good ability to detect object boundaries from feature images and was trained based on both normal and quality problem samples, enabling the final model to extract window and door boundaries with good accuracy from point cloud data containing some occlusion and misalignment problems. Compared with traditional facade map extraction methods such as slicing-based methods, the proposed method can manage data problems more effectively. Figure 19 shows the results of the point cloud data with misalignment. Although the misalignment problem is serious in some areas of the three facades, the proposed method can still obtain window boundaries with high quality.

Table 5. Comparison of the two methods' extraction results with the IQmulus & TerraMobilita Contest dataset.

Results by VCIR Method	Results by Proposed Method
I10	I7, I8, I9
I11	I4, I5, I6
I12	I2, I3
I13	I0, I1
I14	I10, I11, I12

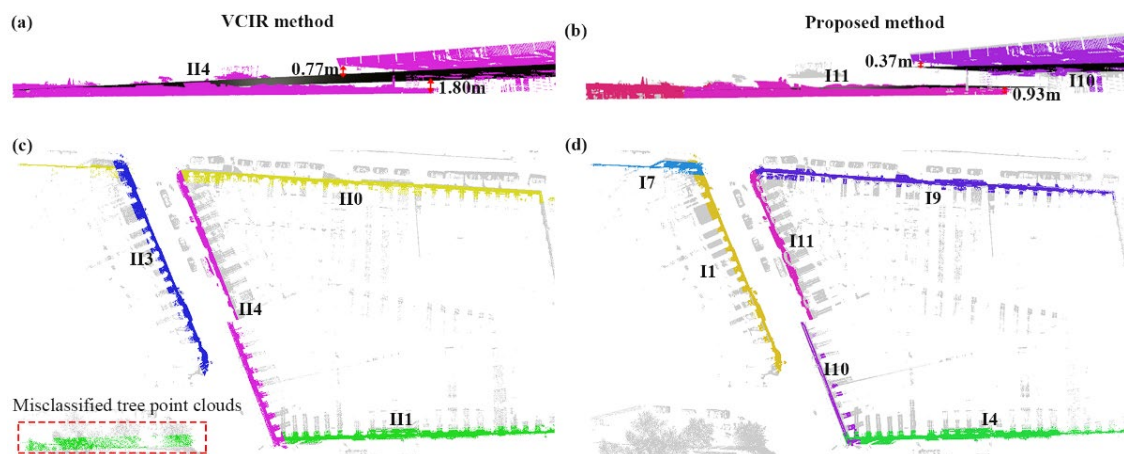


Figure 18. Details of planes extracted by the VCIR method (a,c) and the details of facades extracted by the proposed method (b,d): (a,b) the details of plane II4 extracted by the VCIR method corresponding to the facades I10 and I11 extracted by the proposed method; and (c,d) the details of plane II1 extracted by the VCIR method corresponding to the facade I4 extracted by the proposed method.



Figure 19. Details of facade map extraction: (a,c,e) results of the facade map extraction for facades I11, C3, and C5, respectively; and (b,d,f) results of zooming in on the misaligned area.

In addition, to evaluate the robustness of the proposed method, we experimented with two different datasets: the IQmulus & TerraMobilita Contest dataset, and the Seman-

tic3D.Net Benchmark dataset. These two datasets were 3D point cloud data collected using the MLS and TLS approaches, respectively, with different data acquisition principles and point cloud characteristics such as density and distribution. Additionally, the two datasets correspond to different cities and scenes, and even the styles of the buildings between the two datasets are remarkably different. Experimental results show that the proposed method can achieve good results on both datasets, highlighting the good data adaptability and robustness of the proposed method. The accuracy of the MLS dataset's results is better than the TLS dataset's results' in both facade extraction and facade map extraction due to the acquisition principle of both methods. The MLS approach constantly moves and scans, acquiring data in more views than the TLS approach, which only acquires data at a few sites. Thus, the MLS dataset has fewer occlusion problems than the TLS dataset. Concurrently, the TLS approach only collects data at a few sites, leading to large differences in density within the point cloud data, while the MLS approach has less variation in overall density due to its mobile scanning. Therefore, compared with the TLS point cloud data, the MLS point cloud data is more suitable for building facade map extraction.

Despite these successes, the proposed method still has some limitations. First, although the shift vote improves the efficiency and memory expenditure of the 3D HT method, it is still inadequate due to the shortcomings of the GHT method itself. More efficient HT algorithms such as the Kernel-based Hough Transform method [15] should likely replace the GHT method in the future. Second, although many methods are used in this study to improve the accuracy of facade map extraction, it is difficult to manage irregularly shaped windows and doors. This issue occurs because the Faster R-CNN model is primarily applicable to rectangular objects. In particular, the TLS dataset contains more arches and irregular windows, which reduces its window and door boundary extraction accuracy. In the future, more object detection models, such as Mask R-CNN, should be used to improve the extraction accuracy of irregular window and door boundaries.

Due to the complexity of the real world and various problems in the acquisition and processing of point cloud data, building facade map extraction based on point cloud data still has many challenges. However, there are still considerable advantages in the efficiency and cost of automatically extracting building facade maps based on point cloud data compared to traditional manual measurements. The facade maps extracted by the proposed method can be used in real production with a small amount of manual correction and can be helpful for urban modeling and planning.

5. Conclusions

An automatic and robust method to accurately extract building facade maps can be applied to urban old city reconstruction and urban planning and may serve to reconstruct large-scale 3D building models. This paper proposes a new method to extract building facade maps automatically and robustly based on 3D point cloud data. The entire process of the proposed method is divided into two steps: building facade extraction and building facade map extraction. In building facade extraction, we first improve the 3D HT algorithm to alleviate the peak fuzziness and dependence on step size selection of the traditional 3D HT algorithm using shift vote and 3D convolution with the accumulator. Then, we combine various algorithms, such as RANSAC and HDBSCAN, to differentiate adjacent facades and accurately extract facade point clouds. For building facade map extraction, we combine the Faster R-CNN model in deep learning image object detection and digital image processing techniques to achieve facade map extraction robustly and precisely. With the input of 3D point cloud data, the proposed method can automatically generate the facade maps of each facade.

The proposed method was evaluated on the IQmulus & TerraMobilita Contest dataset and the Semantic3D.Net Benchmark dataset, which were obtained using the MLS and TLS approaches, respectively. For the IQmulus & TerraMobilita Contest dataset, the total MAE and MSE of the extracted building facade are less than 0.32 m and 0.2 m, which is only approximately half of the corresponding error of the VCIR method (0.55 m and 0.44 m,

respectively). The average MAE and MSE for a single facade are less than 0.41 m and 0.28 m, respectively, while the corresponding accuracy indices of the VCIR method are 0.56 m and 0.44 m, respectively. With the Semantic3D.Net Benchmark dataset, the total MAE and MSE of the extracted building facade are less than 0.34 and 0.23 m, which is only approximately half of the corresponding error of the VCIR method (0.5 m and 0.42 m, respectively). The average MAE and MSE for a single facade are less than 0.41 m and 0.28 m, respectively, while the corresponding accuracy indices of the VCIR method are 0.69 m and 0.64 m, respectively. These results indicate that facade extraction accuracy is markedly higher with the proposed method. In building facade map extraction, for the IQmulus & TerraMobilita Contest dataset, the minimum and average AP_{50} of window boundary extraction reach 0.938 and 0.976, and the minimum and average AP_{85} reach 0.621 and 0.827. With the Semantic3D.Net Benchmark dataset, the average AP_{50} and AP_{85} of window boundary extraction are all over 0.86 and 0.77, which means windows can be extracted correctly and the method can obtain an accurate facade map for most facades.

In this study, we present a new, robust method that can automatically extract accurate building facade point clouds and vectorized facade maps from point cloud data with uneven density, noise, and occlusion, and does not require auxiliary information such as point cloud intensity and color. The method's robustness has been validated with two datasets, which were collected using different approaches from different cities with different building styles. It is a beneficial attempt for point cloud information extraction and building 3D reconstruction. Although shift vote is used to improve the efficiency of the proposed method on facade extraction, the method is still limited by the inefficiency of the GHT method itself. In the future, the efficiency of the potential plane acquisition could be improved with higher performance HT methods, such as Kernel-based Hough Transform. Moreover, the deep learning image segmentation method will be considered to identify building edges to improve the accuracy of building edge extraction.

Author Contributions: Funding acquisition, B.Y. and X.D.; Methodology, J.H. and B.Y.; Supervision, X.D.; Validation, D.X., T.W., Y.H. and B.W.; Visualization, J.H. and B.Z.; Writing—original draft, J.H.; Writing—review & editing, B.Y., X.D. and K.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was jointly supported by the National Natural Science Foundation of China (grant numbers: 41941019, 42072306), the Youth Science Fund of the National Natural Science Foundation of China (grant numbers: 41801399), the China Postdoctoral Science Foundation (grant numbers: 2019M653476), and the Young Teachers "Passing Academic Barriers" Funding Program of Southwest Petroleum University (grant numbers: 201599010140).

Acknowledgments: The authors appreciate the IQmulus & TerraMobilita Contest for the free deliveries of the point cloud data. Constructive comments from the anonymous reviewers are also greatly appreciated.

Conflicts of Interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Wang, Y.; Ma, Y.; Zhu, A.; Zhao, H.; Liao, L. Accurate Facade Feature Extraction Method for Buildings from Three-Dimensional Point Cloud Data Considering Structural Information. *ISPRS J. Photogramm. Remote Sens.* **2018**, *139*, 146–153. [\[CrossRef\]](#)
2. Liang, X.; Fu, Z.; Sun, C.; Hu, Y. MHIBS-Net: Multiscale Hierarchical Network for Indoor Building Structure Point Clouds Semantic Segmentation. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *102*, 102449. [\[CrossRef\]](#)
3. Wang, Q.; Kim, M.-K. Applications of 3D Point Cloud Data in the Construction Industry: A Fifteen-Year Review from 2004 to 2018. *Adv. Eng. Inform.* **2019**, *39*, 306–319. [\[CrossRef\]](#)
4. Malihi, S.; Valadan Zoej, M.J.; Hahn, M. Large-Scale Accurate Reconstruction of Buildings Employing Point Clouds Generated from UAV Imagery. *Remote Sens.* **2018**, *10*, 1148. [\[CrossRef\]](#)
5. Teboul, O.; Simon, L.; Koutsourakis, P.; Paragios, N. Segmentation of building facades using procedural shape priors. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 3105–3112.

6. Xie, L.; Zhu, Q.; Hu, H.; Wu, B.; Li, Y.; Zhang, Y.; Zhong, R. Hierarchical Regularization of Building Boundaries in Noisy Aerial Laser Scanning and Photogrammetric Point Clouds. *Remote Sens.* **2018**, *10*, 1996. [\[CrossRef\]](#)
7. Xie, L.; Hu, H.; Zhu, Q.; Li, X.; Tang, S.; Li, Y.; Guo, R.; Zhang, Y.; Wang, W. Combined Rule-Based and Hypothesis-Based Method for Building Model Reconstruction from Photogrammetric Point Clouds. *Remote Sens.* **2021**, *13*, 1107. [\[CrossRef\]](#)
8. Zhou, M.; Ma, L.; Li, Y.; Li, J. Extraction of building windows from mobile laser scanning point clouds. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 4304–4307.
9. Hao, W.; Wang, Y.; Liang, W. Slice-Based Building Facade Reconstruction from 3D Point Clouds. *Int. J. Remote Sens.* **2018**, *39*, 6587–6606. [\[CrossRef\]](#)
10. Li, J.; Xiong, B.; Biljecki, F.; Schrotter, G. A Sliding Window Method for Detecting Corners of Openings from Terrestrial LiDAR Data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *42*, 97–103. [\[CrossRef\]](#)
11. Zolanvari, S.I.; Laefer, D.F. Slicing Method for Curved Facade and Window Extraction from Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2016**, *119*, 334–346. [\[CrossRef\]](#)
12. Biosca, J.M.; Lerma, J.L. Unsupervised Robust Planar Segmentation of Terrestrial Laser Scanner Point Clouds Based on Fuzzy Clustering Methods. *ISPRS J. Photogramm. Remote Sens.* **2008**, *63*, 84–98. [\[CrossRef\]](#)
13. Dong, Z.; Yang, B.; Hu, P.; Scherer, S. An Efficient Global Energy Optimization Approach for Robust 3D Plane Segmentation of Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2018**, *137*, 112–133. [\[CrossRef\]](#)
14. Maas, H.-G.; Vosselman, G. Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data. *ISPRS J. Photogramm. Remote Sens.* **1999**, *54*, 153–163. [\[CrossRef\]](#)
15. Limberger, F.A.; Oliveira, M.M. Real-Time Detection of Planar Regions in Unorganized Point Clouds. *Pattern Recognit.* **2015**, *48*, 2043–2053. [\[CrossRef\]](#)
16. Xu, Y.; Ye, Z.; Huang, R.; Hoegner, L.; Stilla, U. Robust Segmentation and Localization of Structural Planes from Photogrammetric Point Clouds in Construction Sites. *Autom. Constr.* **2020**, *117*, 103206. [\[CrossRef\]](#)
17. Adam, A.; Chatzilari, E.; Nikolopoulos, S.; Kompatsiaris, I. H-RANSAC: A Hybrid Point Cloud Segmentation Combining 2D and 3D Data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *4*, 1–8. [\[CrossRef\]](#)
18. Ebrahimi, A.; Czarnuch, S. Automatic Super-Surface Removal in Complex 3D Indoor Environments Using Iterative Region-Based RANSAC. *Sensors* **2021**, *21*, 3724. [\[CrossRef\]](#)
19. 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 (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 77–85.
20. Lin, H.; Wu, S.; Chen, Y.; Li, W.; Luo, Z.; Guo, Y.; Wang, C.; Li, J. Semantic Segmentation of 3D Indoor LiDAR Point Clouds through Feature Pyramid Architecture Search. *ISPRS J. Photogramm. Remote Sens.* **2021**, *177*, 279–290. [\[CrossRef\]](#)
21. Chen, Y.; Wu, R.; Yang, C.; Lin, Y. Urban Vegetation Segmentation Using Terrestrial LiDAR Point Clouds Based on Point Non-Local Means Network. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *105*, 102580. [\[CrossRef\]](#)
22. Haghighatgou, N.; Daniel, S.; Badard, T. A Method for Automatic Identification of Openings in Buildings Facades Based on Mobile LiDAR Point Clouds for Assessing Impacts of Floodings. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *108*, 102757. [\[CrossRef\]](#)
23. Alshawabkeh, Y. Linear Feature Extraction from Point Cloud Using Color Information. *Herit. Sci.* **2020**, *8*, 28. [\[CrossRef\]](#)
24. Díaz-Vilariño, L.; Khoshelham, K.; Martínez-Sánchez, J.; Arias, P. 3D Modeling of Building Indoor Spaces and Closed Doors from Imagery and Point Clouds. *Sensors* **2015**, *15*, 3491–3512. [\[CrossRef\]](#)
25. Duda, R.O.; Hart, P.E. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Commun. ACM* **1972**, *15*, 11–15. [\[CrossRef\]](#)
26. Borrmann, D.; Elseberg, J.; Lingemann, K.; Nüchter, A. The 3D Hough Transform for Plane Detection in Point Clouds: A Review and a New Accumulator Design. *3D Res.* **2011**, *2*, 3. [\[CrossRef\]](#)
27. Li, N.; Ma, Y.; Yang, Y.; Gao, S. An Improved Method of Lee Refined Polarized Filter. *Sci. Surv. Mapp.* **2011**, *36*, 144–145+138.
28. Campello, R.J.G.B.; Moulavi, D.; Sander, J. Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 160–172.
29. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 580–587.
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [\[CrossRef\]](#)
31. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot MultiBox detector. In *Computer Vision—ECCV 2016*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 9905, pp. 21–37. [\[CrossRef\]](#)
32. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. *arXiv* **2016**, arXiv:1506.02640.
33. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *arXiv* **2016**, arXiv:1506.01497. [\[CrossRef\]](#)
34. Ma, L.; Liu, Y.; Zhang, X.; Ye, Y.; Yin, G.; Johnson, B.A. Deep Learning in Remote Sensing Applications: A Meta-Analysis and Review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *152*, 166–177. [\[CrossRef\]](#)

35. Esri Automation of Map Generalization: The Cutting-Edge Technology. 1996. White Paper. Redlands, ESRI Inc. Available online: http://downloads.esri.com/support/whitepapers/ao_/mapgen.pdf (accessed on 12 June 2022).
36. Vallet, B.; Brédif, M.; Serna, A.; Marcotegui, B.; Paparoditis, N. TerraMobilita/IQmulus Urban Point Cloud Analysis Benchmark. *Comput. Graph.* **2015**, *49*, 126–133. [[CrossRef](#)]
37. Hackel, T.; Savinov, N.; Ladicky, L.; Wegner, J.D.; Schindler, K.; Pollefeys, M. Semantic3d. Net: A New Large-Scale Point Cloud Classification Benchmark. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *41*, 91–98. [[CrossRef](#)]
38. Smith, L.N. Cyclical learning rates for training neural networks. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 464–472.
39. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Neural Information Processing Systems (NIPS): La Jolla, CA, USA, 2017; Volume 30.