A Hierarchical Methodology for Urban Facade Parsing from TLS Point Clouds

Zhuqiang Li[1], Liqiang Zhang[1,*], P. Takis Mathiopoulos[2], Fangyu Liu[1], Liang Zhang[1], Shuaipeng Li[1], Hao Liu[1]

[1]*State Key Laboratory of Remote Sensing Science, Beijing Normal University, Beijing 100875, China*

[2]*Department of Informatics and Telecommunications, National and Kapodestrian University of Athens, Athens 157 84, Greece*

Abstract: The effective and automated parsing of building facades from terrestrial laser scanning (TLS) point clouds of urban environments is an important research topic in the GIS and remote sensing fields. It is also challenging because of the complexity and great variety of the available 3D building facade layouts as well as the noise and data missing of the input TLS point clouds. In this paper, we introduce a novel methodology for the accurate and computationally efficient parsing of urban building facades from TLS point clouds. The main novelty of the proposed methodology is that it is a systematic and hierarchical approach that considers, in an adaptive way, the semantic and underlying structures of the urban facades for segmentation and subsequent accurate modeling. Firstly, the available input point cloud is decomposed into depth planes based on a data-driven method; such layer decomposition enables similarity detection in each depth plane layer. Secondly, the labeling of the facade elements is performed using the SVM classifier in combination with our proposed BieS-ScSPM algorithm. The labeling outcome is then augmented with weak architectural knowledge. Thirdly, least-squares fitted normalized gray accumulative curves are applied to detect regular structures, and a binarization dilation extraction algorithm is used to partition facade elements. A dynamic line-by-line division is further applied to extract the boundaries of the elements. The 3D geometrical façade models are then reconstructed by optimizing facade elements across depth plane layers. We have evaluated the performance of the proposed method using several TLS facade datasets. Qualitative and quantitative performance comparisons with several other state-of-the-art methods dealing with the same facade parsing problem have demonstrated its superiority in performance and its effectiveness in improving segmentation accuracy.

Keywords: TLS point cloud, Facade parsing, Layer decomposition, Semantic segmentation, 3D modeling.

1. Introduction

Accurate representation of 3D facade elements has become very important for urban building modeling because such models have a wide range of applications in GIScience and remote sensing fields, including urban reconstruction (Zhang *et al*., 2013; Sampath and Shan, 2010) and damage assessment (Kemec *et al*., 2010). Recent advances in terrestrial laser scanning (TLS) devices allow quick acquisition of unorganized point clouds of large-scale urban scenes (Zhu and Zhang, 2014; Chen *et al*., 2014). Various techniques that make use of TLS point clouds for building detection and reconstruction have been developed (Riemenschneider *et al*., 2012; Musialski *et al*., 2012; Martinovic *et al*., 2015; Zheng *et al*., 2010; Friedman and Stamos, 2013; Friedman and Stamos, 2013; Wan and Sharf, 2012; Hohmann *et al*., 2009; Hohmann *et al*., 2010; Nan *et al*., 2010; Vanegas *et al*., 2012; Zhou and Neumanu, 2011; Toshev *et al*., 2010; Pu and Vosselman, 2009; Brenner, 2005; Lafarge *et al*., 2008; Ripperda and Brenner, 2009; Li *et al*., 2011; Zhang *et al*., 2013). However, automatically parsing urban building facades from unorganized TLS point clouds still remains a challenging research topic for several reasons. For example, building layouts often have high diversity, while on the other hand, occlusions are omnipresent in urban scenes. Lower parts of facades are often occluded by many types of vegetation, street signs, cars and pedestrians. As a result, the obtained point clouds typically suffer from large amounts of missing data because of occlusion as well as from uneven point densities.

The focus of this paper is on 3D building facades modeling. The scientific literature on this topic can be broadly divided into three categories: *i*) Image-based, *ii*) Point cloud-based, and *iii*) Fusion based. The next two subsections will be an overview of the known techniques of the last two categories owing to its direct relationship with the subject of this paper.

1.1. Point Cloud-based Methods

To reconstruct building facades from noisy TLS point clouds, Riemenschneider *et al*. (2012) exploited repetitions and self-similarities of the building facades to consolidate the imperfect data and complete the missing parts. For each of the facade elements, the planes are detected via the RANSAC algorithm, and the individual elements are registered at a per-plane level. Zheng *et al*. (2010) presented a method for consolidating 3D urban point clouds with poor quality. A drawback of this method is the strict requirement of user interactions. Friedman and Stamos (2013) presented a line sweep algorithm for detecting major planes and repetitive architectural features. However, such a method is unable to effectively handle facades with many structural details. Wan and Sharf (2012) first consolidated the point cloud using the algorithm reported in Tyleček and Šára (2010) and later segmented it into depth planes. Thus, every depth plane is segmented into facades by the optimal sequence of grammar derivations. At each depth plane, the facades are extruded to obtain a polygonal building model. However, this approach suffers from the lack of geometrical details because the shape grammar has difficulty identifying the boundaries of complex structures. Hohmann *et al*. (2009 and 2010) developed a modeling system by combining procedural modeling with GML shape grammars. Although this interactive approach is performed in a top-down and coarse-to-fine manner, it contains high-level cues and aims at semantic enrichment of the geometric models. Based on the concept of semi-automatic snapping of small structural assemblies, Nan *et al*. (2010) modeled high-rise building facades with many repetitive structural elements. In their method, the geometrical parametric primitives are selected to effectively scan facades by interactive user guidance and then are merged and refined to form the final building models. However, their method is only adapted to modeling symmetrical and repetitive building facades. Vanegas *et al*. (2012) proposed another approach for reconstructing buildings from point clouds by considering the assumption of the Manhattan world building geometry. Their proposed system detects and classifies features in the data and organizes them into a connected set of clusters from which a volumetric model description is extracted. However, the primary limitation of their method is that it is unable to handle slanted surfaces. For irregular and complex building modeling, Zhou and Neumann (2011) generated 2.5D polygonal models with arbitrarily detailed shaped roofs that are comparable to those of interactively created models. To find topological structures of building facades, like windows and doors, Martinovic *et al*. (2015) proposed a procedural modelling approach for semantic segmentation of 3D city models by point cloud analysis. They performed semantic classification and splitting process in 3D, obviating the need for slow image based semantic segmentation methods. Structural principles such as symmetries are also combined into 3D labeling to further enhance the performance. The facade splitting quality is dependent on the classification quality, thus the poor point cloud classification results often cause façade over-segmentation. Toshev *et al*. (2010) detected and parsed buildings from unorganized range data by using a dependency parsing algorithm. The output of this algorithm is a set of parse trees, each of which represents a semantic decomposition of a building. Nonetheless, this algorithm has difficulty handling building facades with complex structures and significant occlusions. To efficiently obtain high-quality facade models, model-driven methods were introduced to reconstruct buildings from the point cloud with or without line segments (Brenner, 2005; Lafarge *et al*., 2008). These methods can, in general, handle missing data or sparse regions; however, the models reconstructed by such methods usually differ from their corresponding real world objects. In an effort to improve this performance, Pu and Vosselman (2009) proposed a knowledge-driven approach to fit detailed polygonal facade elements. Specifically, they applied available knowledge about sizes, positions, orientations and topologies of the facade elements in recognizing the elements in a point cloud. However, this approach fails to correctly separate the outlines of complex building structures, such as curved walls and curved protrusions or non-vertical walls.

1.2 Fusion of Image- and Point Cloud-Based Facade Parsing Methods

To enhance TLS point cloud data, You and Lin (2010) fused the data with topographic maps so that more accurate building models could be constructed. Although their approach can obtain accurate details of the structures in the horizontal direction, it is inaccurate in the vertical direction. A structural description of a facade was derived from range and image data by a reversible-jump Markov Chain Monte Carlo approach (Ripperda and Brenner, 2009). In this work, although a context-free grammar for facades was derived from a set of facade images and fitted to new models, the particular method was not appropriate for enriching buildings with free-form facades. Combining the advantages of 3D LiDAR scans and 2D images, Li *et al*. (2011) presented a method for the layer decomposition of urban facades. By solving a multi-label assignment problem, they decomposed the 2D facade image into rectangular planar fragments and transferred diffuse depth information from the corresponding 3D scan onto the fragments. Unfortunately, although the layer decomposition can produce an enhanced, layered, and textured model, data sources such as topographic maps are not always available. Many state-of-the-art methods use shape grammars as higher-order knowledge models for the automated parsing of building facades by assuming that the appropriate rules are available from the outset (Teboul *et al*., 2013; Teboul *et al*., 2011; Koziński *et al*., 2015; Riemenschneider *et al*., 2012; Zhang and Wu *et al*., 2014). For example, Teboul *et al*. (2013) used the binary split grammars to partition a building façade image into a set of predefined semantic classes. Then reinforcement learning techniques are applied to optimize the parsing results. Nonetheless, designing a known set of grammar rules is difficult and limits the range of the method's applicability. Moreover, creating style-specific grammars is a time-consuming process (Martinovic and Van, 2013). To reduce the complexity of recognizing facade elements, a three-layer approach for the semantic segmentation of building facades has been proposed (Martinovic *et al.*, 2012). In particular, a machine learning technique was used based upon a recursive neural network (RNN) (Socher *et al.*, 2011) coupled with information obtained by specialized facade component detectors. The raw RNN output data were combined with information from object detectors trained to detect architectural elements. There are several shortcomings of this method, including the fact that the RNN system requires a high number of training samples and thus is computationally intensive. Moreover, the additional cost of manually annotating training data, i.e. by individual users, is usually very high both in terms of complexity and time execution.

1.3. Contributions

Taking TLS point clouds which are inherently 3D into consideration, we have developed, analyzed and evaluated the performance of a novel methodology based upon a hierarchical framework, for the effective 3D parsing of urban building facades. The methodology is implemented using the following hierarchical operational procedures: *i*) Decomposition of prototyping elements, *ii*) Semantic segmentation, and *iii*) Parsing optimization and 3D modeling. High-quality 3D building facade models were reconstructed from the input point cloud, even in cases where the available input data were contaminated with severe missing areas or were corrupted by noise and outliers.

The specific contributions of this paper can be summarized as follows:

i) The development of a novel general framework for building facade parsing in which the classification results combined with a priori semantic knowledge improve the labeling accuracy of the facade elements. This generic framework has the ability to adaptively segment and accurately model facades of various complexities and styles.

ii) The application of a novel Bicubic Interpolation Extending Sample (BieS) algorithm which has the property to make the number of the segmented regions for each class be the same. This ensures that the extracted features using the ScSPM from the data samples are discriminative.

iii) The introduction and analysis of a morphological boundary detection and dynamic split optimization method ensuring obtainment of accurate positions and ranges of facade elements. In this way, the precise element boundaries are also easily derived, which allows us to generate high-quality and accurate 3D building facade models.

2. Overview of the Methodology

The flowchart of the proposed methodology is illustrated in Fig. 1 as a generic block diagram. The methodology consists of the following functionalities: Decomposition (see Section 3); Semantic segmentation (see Section 4); and Optimization and 3D modeling (see Section 5). As shown in Fig. 1, a data-driven method is used to decompose the input facade point cloud dataset into 2.5D depth planes. Semantic segmentation and underlying facade structures, such as repetitive and symmetric patterns, are used as priors for parsing the facade elements of each depth plane. The extracted facade elements are optimized and then their boundaries are accurately extracted. Finally, the 3D building facade models are reconstructed by integrating the optimized elements. In the next three sections, the detailed operation of the aforementioned functionalities is presented.

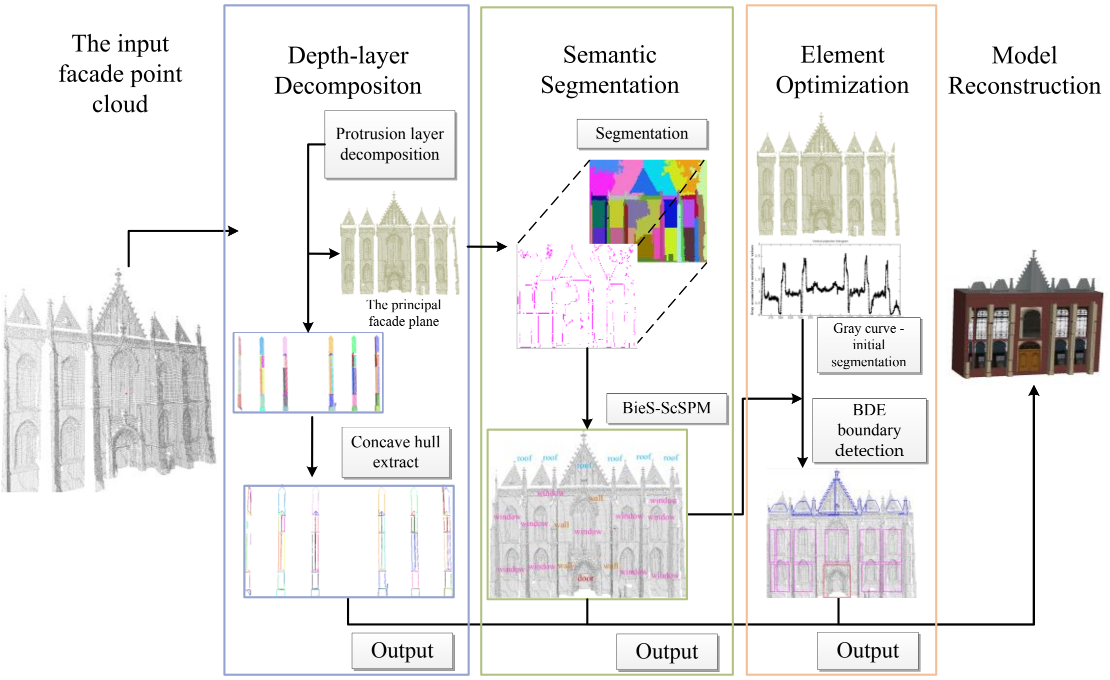


Fig. 1. The block diagram of the proposed methodology.

3. Decomposition of Prototyping Elements

Since building facades often contain protruding elements such as balconies, the point clouds need to be transformed into a structured set of 2.5D depth planes to enable extraction of the boundaries of the facade elements in each layer.

To decompose a facade into depth planes, the point set needs to be segmented into

2.5D segments. To this end, the point cloud is separated by the curvature-based region growing algorithm (Jagannathan and Miller, 2007), which merges the neighboring points and adds a smoothness constraint. In particular, the point with the minimum curvature is first set as a seed and then the angles between the normal of every neighbor and normal of the current seed point are computed. If their angle is less than a certain threshold, the neighboring points are considered to be on the same plane; otherwise, they are declared to be on different planes. In practice, the typical vector range threshold θ_{seg} is set within the range of 4° to 6°. Therefore, the output of this algorithm is a set of point clusters where the point set in each cluster is supposed to be located on the same plane. The plane having the majority of points determines two of the axes and the third is normal to the plane. This plane, which will be referred to as the facade principal plane, is derived from the input point cloud by using the RANSAC algorithm (Schnabel *et al*., 2007; Teeravech *et al*., 2014). In the coordinate system used for layer decomposition, the x-axis is along the horizontal direction of the facade principal plane, the y-axis is along the vertical direction of the facade principal plane, and the z-axis is perpendicular to the facade principal plane; its positive direction points to the outside of the facade. After all the planes in the point clusters are detected by the curvature-based region growing algorithm, the RANSAC algorithm is re-applied to extract the planes from the clusters. Layer decompositions are derived according to the distance between the planes (detected as described above) and the facade principal plane. The distance threshold for segmenting the depth planes can be expressed as

$$\Delta dis\tan ce = mean_{seg} \pm \frac{\varepsilon_{seg}}{2} \tag{1}$$

where $mean_{seg}$ is the average distance between the segmented planes and the principal plane and $\varepsilon_{\mathrm{seg}}$ is the standard error in the distance estimation.

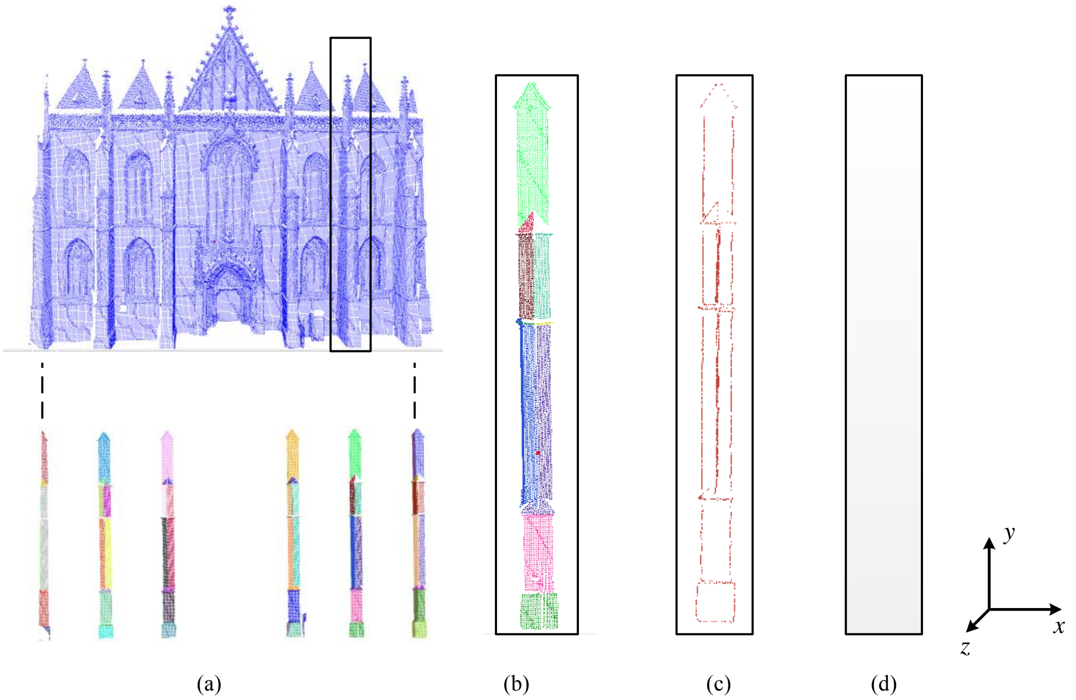


Fig. 2. Decomposition of protruding elements and boundary extraction on Cathedral facade data. (a) The pillars are segmented from the input point cloud (top left). (b) The segmented pillar illustrated in the rectangle of the input point cloud. (c) Boundary points (brown points) of the pillar in (b). (d) The ground truth of the pillar boundary by manual annotation.

The procedure described above converts 3D facade point clouds into 2.5D components. The boundaries of the protruding elements of the building facades can be computed by applying the α-shape technique (Edelsbrunner *et al.*, 1983). As illustrated in Fig. 2, the Cathedral facade point cloud located at the top of Fig. 2(a) is decomposed into multi-layers, whereas the protruding elements are illustrated at the bottom of the same figure with different colors representing different segmented planes. Fig. 2(b) is the zoom-in of a pillar and Fig. 2(c) shows the boundary points of the pillar. Comparing the ground truth data of Fig. 2(d) and the extracted boundary shown in Fig. 2(c) of the pillar, it is clear that the structures of the pillar are complete.

Figs. 3 and 4 illustrate the outcome of the protruding element decomposition. Specifically, Fig. 3(a) shows an apartment point cloud whereas Fig. 3(b) shows the segmented planes derived by the region-growing algorithm, where it can be seen the balconies have been extracted from the input cloud. Fig. 3(c) illustrates the simplified boundaries, and it can be clearly observed from the zoom-in shown in Fig. 3(d) that our method accurately extracts the boundaries of the elements. Fig. 4(a) illustrates a more complex facade point cloud. The dataset shown in Fig. 4(a) has been decomposed into two depth planes, the balcony layer in Fig. 4(b) and the wall layer connected with the balcony as shown in Fig. 4(d).

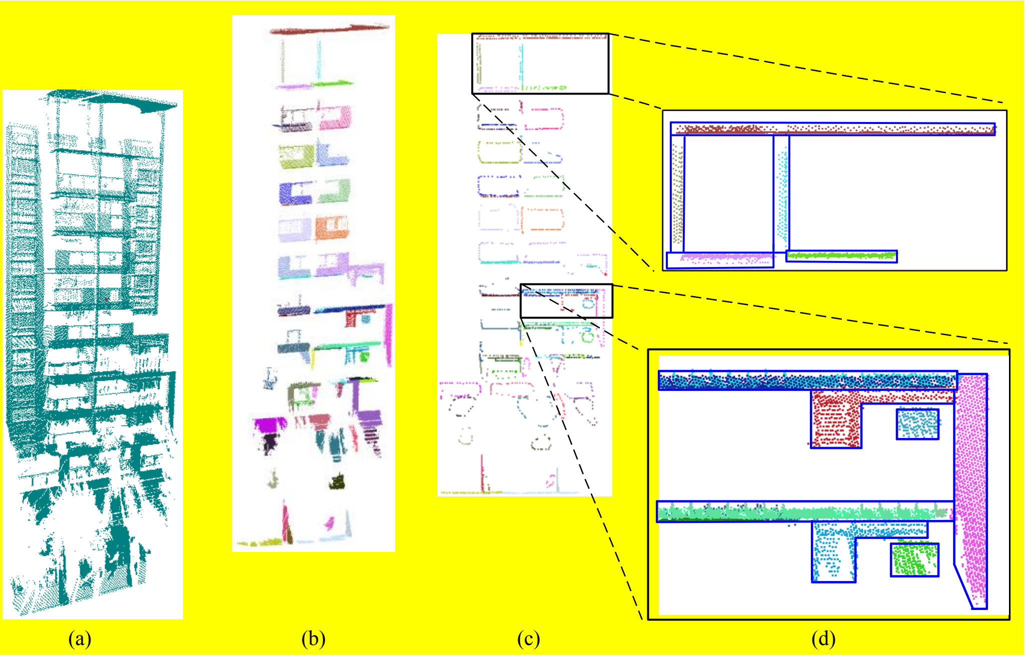


Fig. 3. Decomposition of an urban facade from a noisy point cloud. (a) The original point cloud with noise and missing data. (b) Layer decomposition of the point cloud. (c) The extracted boundaries of the elements. (d) Zoom-ins of the element boundaries which is overlaid with point cloud in the black rectangles.

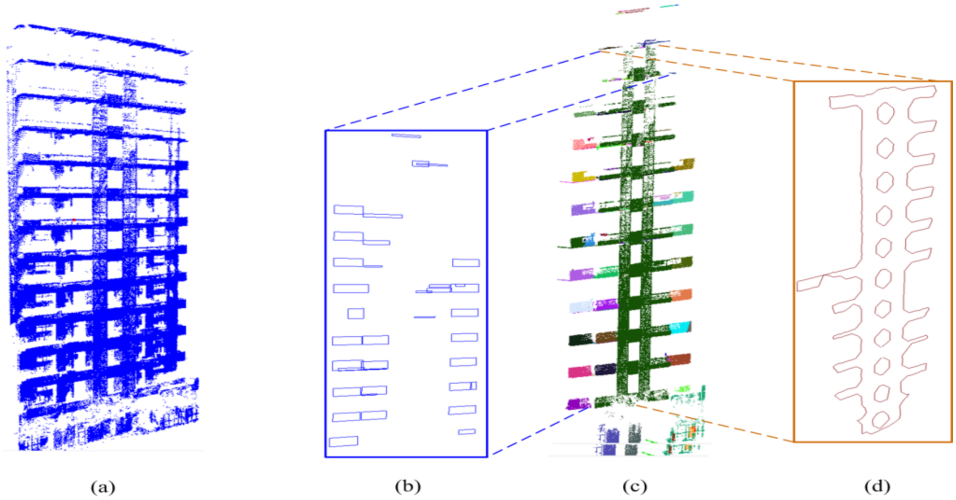


Fig. 4. Protruding element decomposition of a high building facade. (a) Original point cloud. The elements of (b) and (d) are decomposed from (c).

4. Semantic Segmentation

The purpose of this operation is to segment and label the depth planes into semantic classes such as roofs, windows and shops. Fig. 5 presents the main operational procedures for the realization of semantic segmentation in block diagram form. The depth image of the output facade point cloud of Section 3 is first over-segmented using the entropy rate superpixel segmentation method (Liu *et al*., 2011). Then, the dimensional scaling factors of the training and testing samples are computed. The BieS algorithm is used to extend the samples. Next, the ScSPM algorithm (Yang *et al*., 2009) is used to extract the features from the samples. Each superpixel is classified into one semantic class (window, wall, roof, shop or door) using a linear SVM classifier based on the features learned from the training stage. Finally, semantic information of the facade elements is applied to improve the classification results.

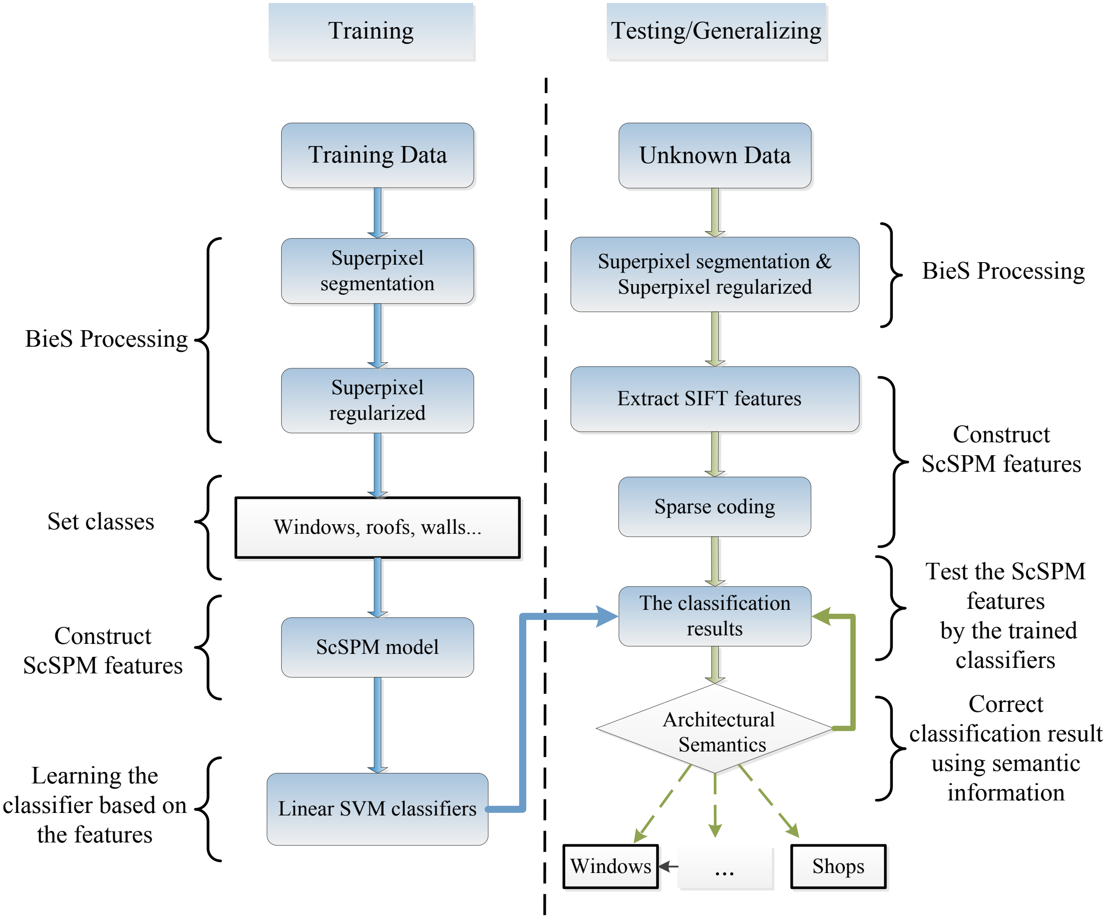


Fig. 5. Overview of the facade semantic classification.

To separate individual facades by detecting differences in their semantic structure, the point cloud of a depth plane is converted into a depth image. The resolution of the depth image depends on the angular precision and range precision of the scanner. The size of the depth image can be obtained by

$$X_g = \frac{X_{\max} - X_{\min}}{pixelsize} + 1 \tag{2}$$

where X_g is the width of the depth image, X_{max} is the maximum in the point cloud along the *X* axis and Y_g and Y_{max} are similarly defined. The gray value of each pixel is computed by

$$gray = \frac{depth_i - depth_{\min}}{depth_{\max} - depth_{\min}} \times 255 \tag{3}$$

where $depth_i$ is the depth value of the *i-th* point, and $depth_{max}$ and $depth_{min}$ are the maximum and minimum depth values of the points, respectively.

4.1. Superpixel Segmentation

To obtain the semantic information of building facades, we used the entropy rate superpixel segmentation method (Liu *et al.*, 2011) to first over-segment the depth image into a set of superpixels.

Assuming that the probability density function of a discrete random variable *M* is given by $p_{M,}$ the entropy of *M* can be mathematically expressed as

$$H(M) = -\sum_{m \in \Gamma} p_M(m) \log p_M(m) \tag{4}$$

where *Γ* is the set of the random variable *M*.

The conditional entropy *H*(*M*|*Y*) quantifies the uncertainty of a random variable *M* conditioned on another random variable *Y* as

$$\begin{aligned} H(M \mid Y) &= \sum_{y \in Y} p_Y(y) H(M \mid Y = y) \\ &= -\sum_{y \in Y} p_Y(y) \sum_{m \in M} p_{X|Y}(m \mid y) \log P_{M|Y}(m \mid y) \end{aligned} \tag{5}$$

In the entropy rate super-pixel segmentation, the larger the entropy rate the more compact the superpixels. In practice, the number of over-segmentations varies between 30 and 50. For example, the Cathedral building facade shown in Fig. 6 is segmented into 35 superpixels. Each superpixel *i* has a label $label_i$ so that all of them form the labeling set, i.e.,

$$\underset{k_1 \times l_1}{label_1} \bigcup \underset{k_2 \times l_2}{label_2} \cdots \bigcup \underset{k_n \times l_n}{label_n} = \underset{k \times l}{labels} \tag{6}$$

$$\underset{k_i \times l_i}{label_i} \bigcap \underset{k_j \times l_j}{label_j} = \varnothing \tag{7}$$

Because the over-segmentation often causes irregularity of the superpixels, the superpixels need to be regularized by the labels obtained by the segmentation. To avoid the case where the extracted features in the generalization of the testing data become inhomogeneous, the length *l* and width *w* of the segmented superpixels are restricted to 30⩽*l*⩽300 and 30⩽*w*⩽300. The final regularized superpixels are used as the training data set (see Section 4.2 for details).

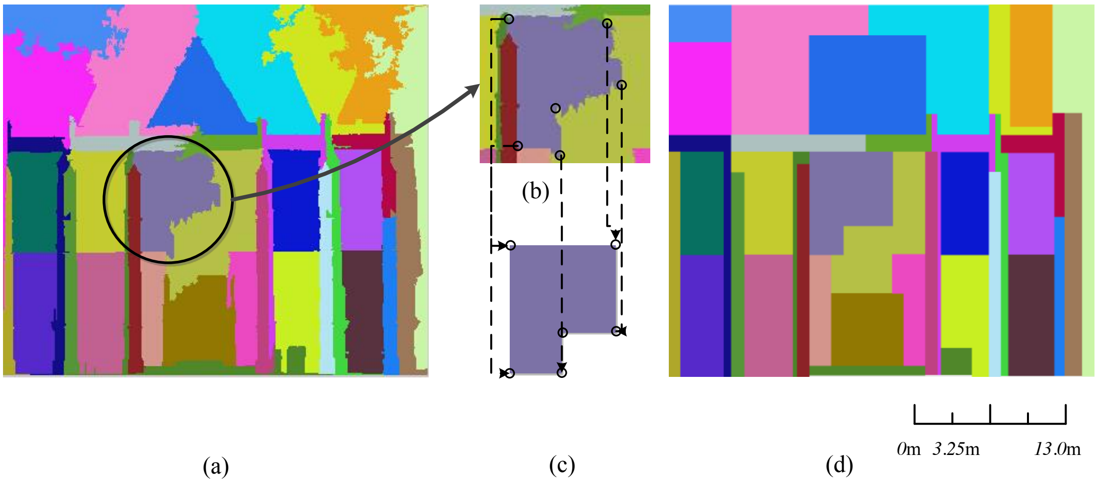


Fig. 6. Regularization of the superpixels. Note that different colors represent different labels. (a) The over-segmented superpixels. (b) A zoom-in of the region in the black circle. (c) Regularization of the region in (b). (d) All regions are regularized.

4.2. Facade Element Labeling

To obtain the semantic information of the superpixels the ScSPM algorithm (Yang *et al.*, 2009) is applied to extract the SIFT features from the 2.5D image under the region determined by each superpixel. The point cloud lacks texture information and the sizes of the facade classes are different because of over-segmentation. Therefore, the SIFT features extracted from the facade depth image are inhomogeneous. To minimize this deficiency we use Eq. (8) to compute the dimensional scaling factors of different training samples. Then, the BieS algorithm is used to extend the training and testing samples.

$$Dsf(i) = \left\| \frac{(\frac{1}{n}\sum_{i=1}^{n} R(i,1)+w)\cdot(\frac{1}{n}\sum_{i=1}^{n} R(i,2)+w)}{R(i,1)\cdot R(i,2)} \right\|^2 \tag{8}$$

$$\underset{n\times 2}{R} = \left\{ \begin{matrix} I_{1row} & I_{1col} \\ I_{2row} & I_{2col} \\ \vdots & \vdots \\ \vdots & \vdots \\ I_{nrow} & I_{ncol} \end{matrix} \right\} \tag{9}$$

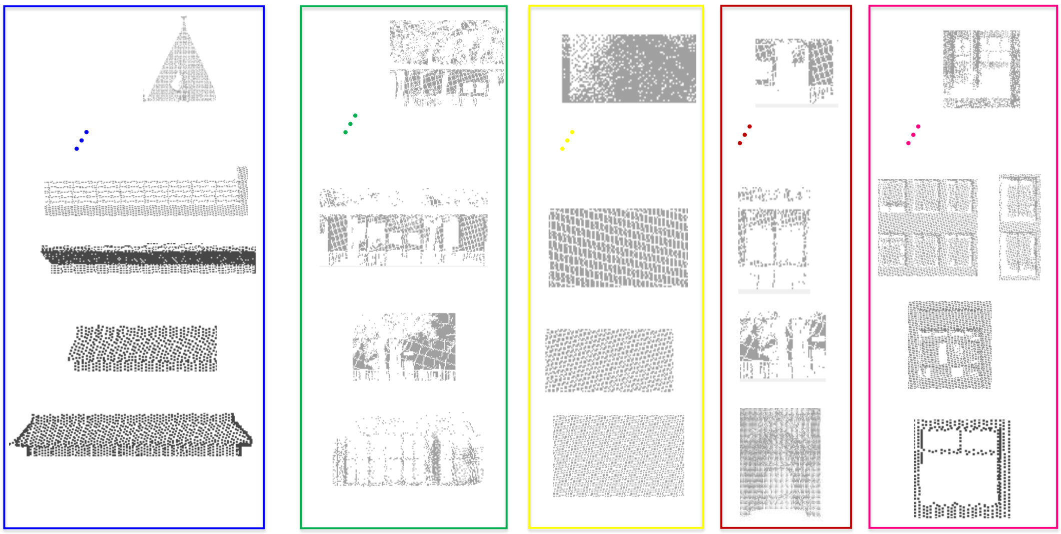
where $\underset{n\times 2}{R}$ is an n×2 matrix and the first and second columns represent the row number and column number of the superpixels corresponding to each class.

Once the scaling coefficients of the samples are computed, the bicubic interpolation

(Eq. 10) is used to resample the samples.

$$I(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3}a_{ij}x^{i}y^{j} \tag{10}$$

An important advantage of using this algorithm is that, for all classes on the facade, the BieS algorithm ensures that the number of the segmented regions for each class is the same. After the SIFT features are extracted by the ScSPM from the above data samples, a linear SVM classifier is trained and then used to classify the testing data. Clearly, there are often some incorrect classification results because of noise and missing data. Therefore, semantic information of the facade elements needs to be integrated into the classification procedure. For example, the windows on a given floor are usually aligned and a balcony is adjacent to the lower part of the window. Moreover, walls are vertical and roofs are either horizontal or sloped and are always located on top of walls. Therefore, if logical errors occur in the semantic information, the second largest probability value in the confusion matrix of the SVM is used as the final classification result.



(a) *Roof* samples

(b) *Shop* samples

(c) *Wall* samples

(d) *Door* samples

(e) *Window* samples

Fig. 7. The training samples for five classes of facade elements: roof, shop, wall, door and window.

An example of how the selection of clean and complete elements in the depth images are selected as the training dataset is shown in Fig. 7. Our training dataset was over-segmented into 82 superpixels used as the training samples of the five classes of facade elements: wall, window, door, roof, and shop. Among the training samples the first four classes required 15 samples, whereas the windows had 22 samples.

Fig. 8 illustrates the classification outcome of the Cathedral building facade point cloud. The facade elements were classified again into the same five classes of facades elements. From Fig. 8(a), we note that the facade classification results had several wrong classes, whereas the classification accuracy obtained by our method was significantly improved, as shown in Fig. 8(b); e.g., the doors were classified correctly.

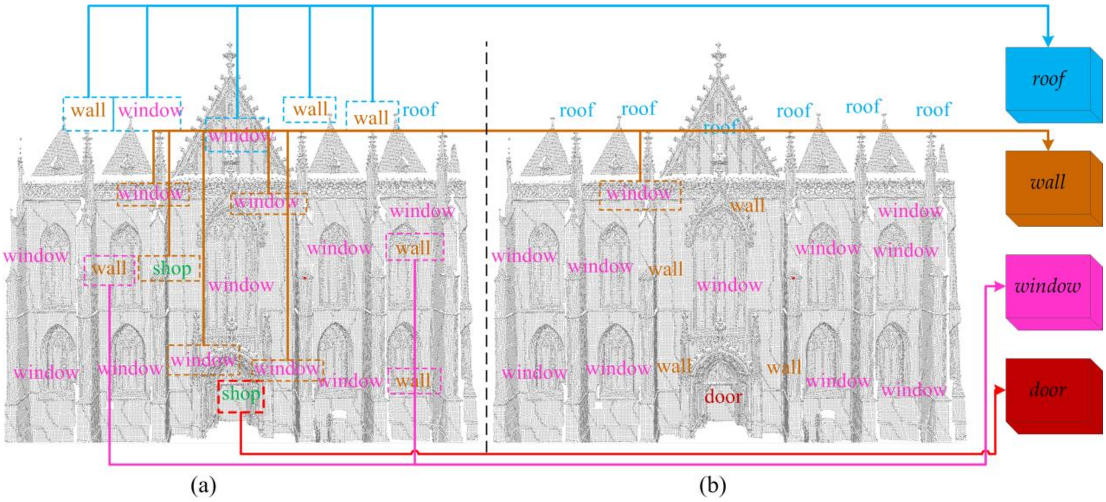


Fig. 8. Cathedral facade classification results. (a) Classification results obtained by the ScSPM. The labels bounded by dashed rectangles are incorrect classification results. (b) Classification results obtained by our method.

5. Parsing Optimization and Modeling

In this section, the extraction and optimization for the element boundaries of the facade elements that leads to the accurate 3D facade reconstruction will be presented. Specifically, the horizontal and vertical gray accumulated curves of the facade depth image are used in Section 5.1 to segment the elements. Because the gray accumulated curves are not smooth and there are many local maxima, the least squares method is employed to fit the curves. The facade is segmented into lattices at the local maxima on horizontal and vertical least squares fitted curves. In Section 5.2, the classification results (obtained as presented in Section 4) are overlaid with the lattices, and the class attribute of each lattice is determined according to the area-dominated principles. In Section 5.3, the morphological boundary detection and the dynamic splitting optimization methods are combined to derive the boundaries of each facade element to enable reconstruction of the 3D facade model.

5.1. Least-squares Fitted Gray Accumulated Normalized Curves

The gray accumulated normalized curve can detect the regular characteristics of the facade such as symmetrical and repeated patterns. Then, these patterns can be used to extract facade elements. The normalized gray accumulated curve Q_x of an $m \times n$ image in the horizontal direction is obtained as follows:

$$Q_i = \frac{\sum_0^m v(i,j)}{\frac{1}{n}\left|\sum_0^m v(1,j) + \sum_0^m v(2,j) + \sum_0^m v(3,j) \cdots + \sum_0^m v(n,j)\right|} \tag{11}$$

where v (i, j) is the gray value at the i-th row and j-th column of the $m \times n$ depth image and Q_i is the normalized gray value of each row i in the image. Similarly, and in an equivalent manner, the normalized gray accumulated curve Q_y in the vertical direction can be derived. The following polynomial expression is used to fit the curves:

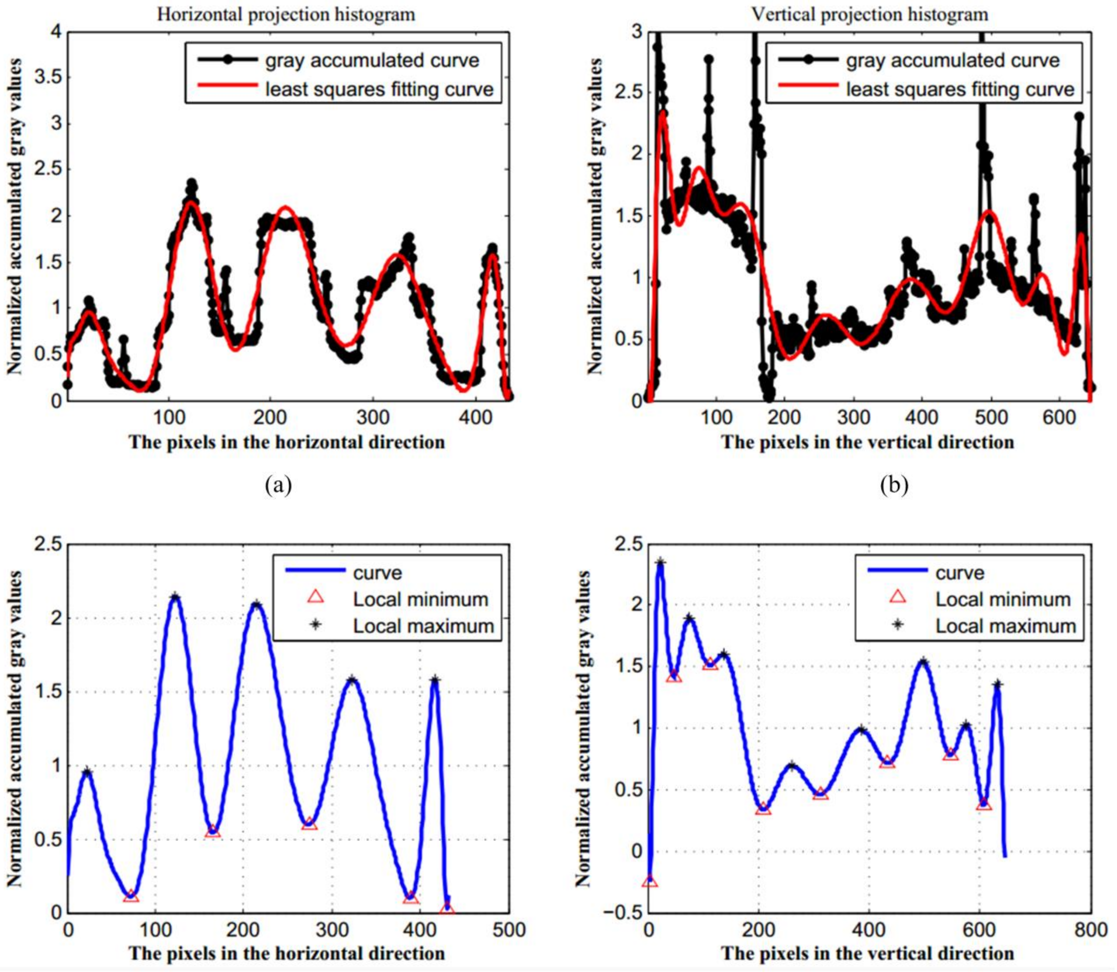
$$Q_i^{'} = a_0 + a_1 x_i + \cdots + a_k x_i^k \tag{12}$$

where x_i is a pixel along the x-axis and $Q_i^{'}$ is the fitted gray value of the i-th row in the depth image by using the least-squares algorithm.

To obtain the parameters of the fitted curves, the quadratic sum of the errors between the fitted curves and their actual gray accumulated curves is first computed. Then these sums are minimized using the least squares method in the horizontal direction as follows:

$$\min R^2 \equiv \sum_{i=1}^{n} (\mathrm{Q}_i - Q_i')^2 \tag{13}$$

Fig. 9 presents the gray accumulated curves and least-squares fitting curves derived from the facade depth image. In Figs. 9 (a) and (b), the black curves denote the normalized gray accumulated curves in the horizontal and vertical directions. Note that the curves had a large fluctuation because of the noise and complex facade structures. In Figs. 9 (c) and (d), the blue curves were fitted by the least-squares method. By this approach, they were in line with the normalized gray accumulative curves and could accurately describe the facade structures. Using Eq. (13), the fitting curve is first determined and then its first-order derivative is computed to obtain the maximum point. As shown in Figs. 9 (c) and (d), the asterisked points were the maximum points that actually were the cut-off points of the structure change. The maximum points of the horizontal fitting curves were connected with those of the vertical fitting curves to generate lattices, and in this way the approximate starting positions of such elements as windows could be determined.



(a)

(b)

(c)

(d)

Fig. 9. The curves derived from the facade depth image. (a) Overlay of the two types of curves in the horizontal direction. (b) Overlay of the two types of curves in the vertical direction. (c) The least-squares fitting curve in the horizontal direction. (d) The least-squares fitting curve in the vertical direction.

5.2. Morphological Boundary Detection

After the lattices with attributes are determined, the facade element boundaries in each lattice can be detected. Although there are no textural features in the depth image, the element edges processed by the binarization operation can be precisely extracted. Because of this, we propose an edge detection method, which will be termed Binarization Dilation Extraction (BDE), to effectively detect the boundaries of each facade element. In this method the data in each lattice are first transformed into a binary image where the conversion threshold is the maximum of the gray level variance. Then, the binary image is dilated with the criterion that the output pixels' values are maximum within their neighborhood of the input image. The structural element *b* is used to dilate the input depth image *f* as follows:

$$(f \oplus b)(x, y) = \max\{f(x-s, y-t) + b(s,t) \big| (x-s, y-t) \in D_f, (s,t) \in D_b\} \qquad (14)$$

where $f(x, y)$ is an input image whose width and length are x and y, respectively. $b(s, t)$ is a structural element that denotes an element of the facade and s and t represent the width and length of the element, respectively.

The boundary effect can be effectively eliminated by filling the missing data regions using the minimum value of the dilation and the maximum value of the erosion operations. The boundary coordinates of each building facade element can be obtained by the detection of the connected component boundary pixels within the binary image, as illustrated in Fig. 10.

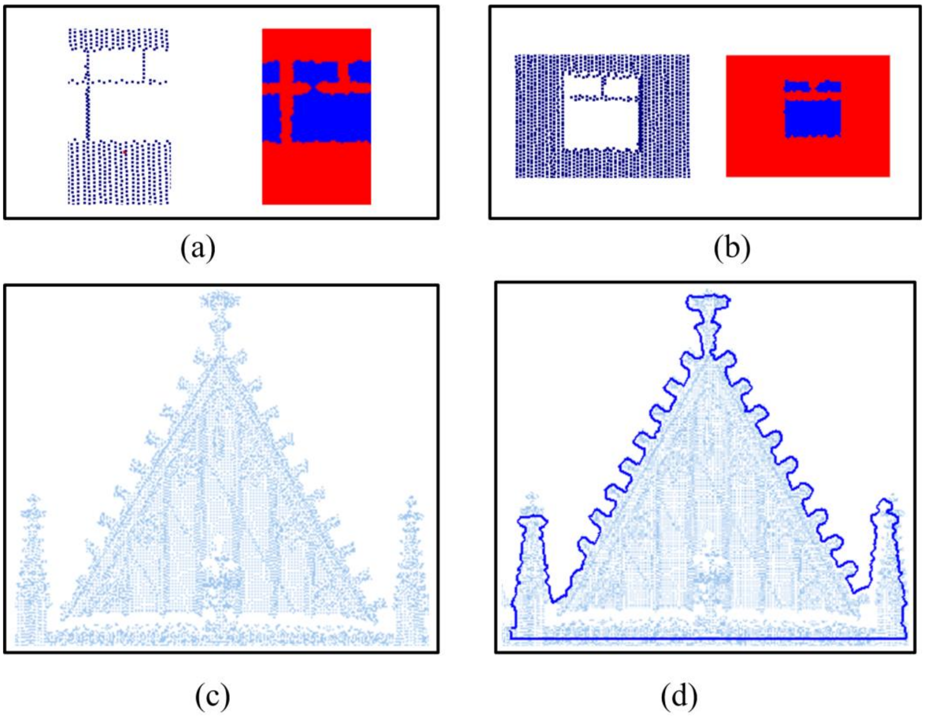
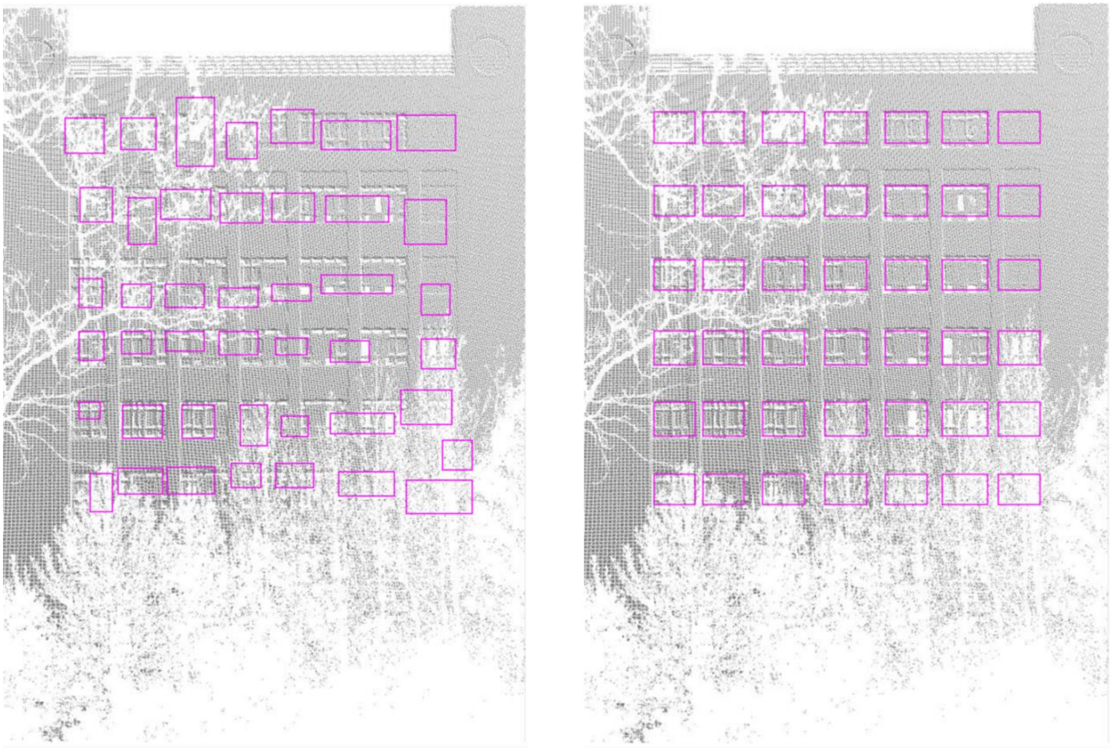


Fig. 10. Boundary extraction of the facade elements using the BDE. In (a) and (b), the left graphs are the depth image and the right graphs are the extracted window regions, colored in blue. (c) The roof depth image. (d) The extracted roof boundary (blue).

5.3. Dynamic Split and Optimization of Local Structures

Although the boundaries of facade elements have been detected as described in the previous subsection, they still contain redundant points. Therefore, we further optimize the obtained elements to remove the redundant points and obtain the correct boundaries. For this purpose, we use the progressive partitioning method to extract the boundaries of the facade elements. First, after the pixels in a row of the depth image are searched, the length, width and starting position of the partition are updated from the previous BDE process. This procedure continues until the ratio between the areas within the extracted boundaries and that of the initial partition region ranges from 0.12 to 0.86. Within this range, the extracted boundaries are considered to be acceptable. The search and partition process is concluded when all pixels of the depth image have been traversed.



(a) (b)

Fig. 11. Extraction and optimization of window boundaries from the incomplete and noisy facade point cloud. (a) Extracted window boundaries. (b) Optimized window boundaries.

The extracted element boundaries are often irregular because of incomplete data. For example, as shown in Fig. 11(a), the boundary sizes vary among the different windows, producing inaccurate positions. To solve this problem, we employ a random interval optimization method to optimize the locations of the extracted facade elements. In particular, the following maximization algorithm is used to optimize the positions and ranges of the windows:

$$\max \sum_{m=1}^{all} \frac{\sum_{y=p_{my+}\Delta p_y}^{y'} \sum_{x=p_{mx}}^{x'} (BDE(f(x,y))==0)}{width_m \times length_m} \qquad (15)$$

where *all* denotes all the windows, $width_m$ is the width of the m-th window, $length_m$ is the length of the m-th window, $f(x, y)$ is the extent of the previously segmented window, $y' = p_{my} + \Delta p_y' + length$, $x' = p_{mx} + width$, p_{my} is the y coordinate of the m-th window and p_{mx} is the y coordinate of the m-th window. Moreover, Δp_y is the distance between two neighboring windows in the vertical direction, which uses values from the following alphabet:

$$\Delta p_y \in \{I_{2,j}(p_y)\text{- } I_{i,j}(p_y),\ldots, I_{t,j}(p_y)\text{- } I_{t-1,j}(p_y)\} \qquad (16)$$

where I is the set of the segmented elements, and $I_{i,j}$ denotes the window at the i-throw and the j-th column from left to right.

The application of this optimization algorithm leads to significant improvements in the positions and sizes of the element boundaries. For example, as it can be seen from Fig. 11, this can be verified by comparing the performance evaluation results. These results have been obtained by using the following procedure. If the window intervals between two neighboring rows are different, $\Delta p_y \in (0, \frac{h_{build} - \min(p_{my}) - g \times \overline{width_m}}{g-1})$, where h_{build} is the building height and g is the building floor number. First, the area bounded by the boundaries extracted by the BDE at different intervals Δp_y is computed. If the ratio between the area within the boundaries extracted by the BDE and the area of the facade element is the maximum, we update the intervals between neighboring elements. This procedure will lead to a more realistic layout of the facade structures, even when incomplete point clouds are being used as input.

To recover the missing structures from the point cloud, a weighted undirected graph is used to organize the point cloud. We denote the graph as $G = (V, E, W)$, where V is the set of the vertices (i.e., the center points of the facade elements), E is the edge set and W is the weight set of the edges. The weight of the edge between the m-th window and the n-th window weight of the edge is given by

$$w_{m,n} = \begin{cases} 0, \quad if \quad m = n \in j \\ \dfrac{1}{\left\| (\overline{p_{mx}}, \overline{p_{my}}), (\overline{p_{nx}}, \overline{p_{ny}}) \right\|^2}, & \text{otherwise} \end{cases} \tag{17}$$

where ($\overline{p_{mx}}$, $\overline{p_{my}}$) are the coordinates of the center point of the m-th window.

Note that the higher $w_{m,n}$ is, the closer the two windows are, and $w_{m,n} = 0$ for $m = n$. After the undirected graph is constructed, the weighted adjacency matrix is used to recover missing structures. We define the operator F_{sim} to be a measure of the similarity between the structures as follows:

$$F_{sim} = \left| \sum_{i=1}^{n} \max(w_{k_i k+1}) \times \overline{p_{k_i y}} - \sum_{i=1}^{s} \max(w_{kk+1_i}) \times \overline{p_{k+1_i y}} \right| \tag{18}$$

where $w_{k_i k+1}$ is the largest weight among the i-th element in the k–th column and all the elements in the (k+1)-th column. $\overline{p_{k_i y}}$ is the y coordinate of the i-th element in the k column.

The smaller the value of F_{sim} becomes, the more similar the elements between the two columns or rows. In Fig. 11, the similarity of the elements among the neighboring columns has been computed and the similarity in the local regions to recover the missing structures has been used. If $F_{sim} \geqslant 2 \Delta F_{sim}$, where ΔF_{sim} is the standard error of F_{sim}, the elements are visually dissimilar. For an occluded element, if only a part of the structure is missing, we determine the similarity with its neighboring element without occlusions where semantic information and the shapes can be used to recover the missing structures of the occluded element. Fig. 12 illustrates the facade element parsing results that have been obtained through the symmetric pattern of the facade. In Fig. 12(a), the windows bounded by yellow rectangles were not detected during the segmentation. The undetected windows were labeled based on the similarity of the elements in the vertical direction computed by Eq. (18). In Fig. 12(b), the windows in the leftmost columns bounded by the red rectangles were less similar compared to the neighboring elements, but they could still be recognized according to the symmetry in the vertical direction. The window in the second column from the left marked by the red point was not segmented initially as shown in Fig. 12(c). It was segmented by similarity as shown in Fig. 12(d), whereas the red element located in the third column was not segmented because of the missing data and dissimilarity with other elements.

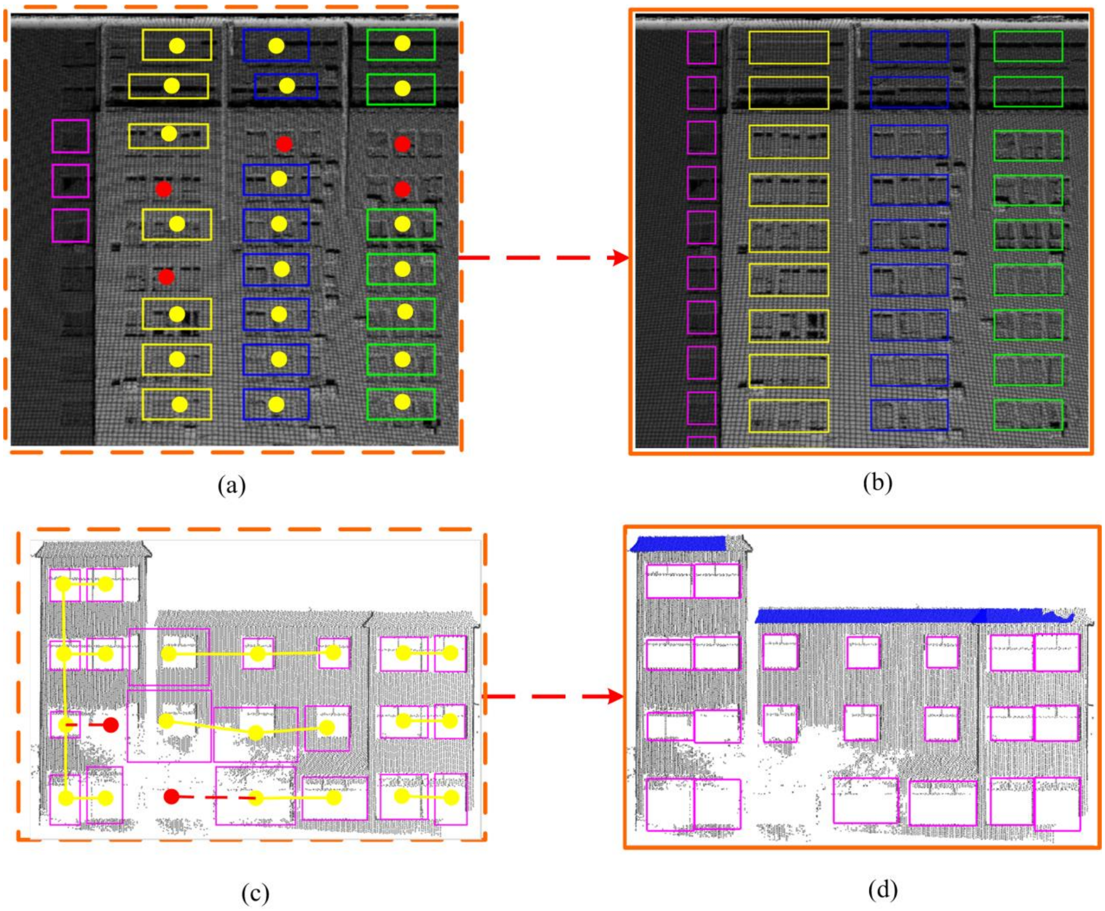
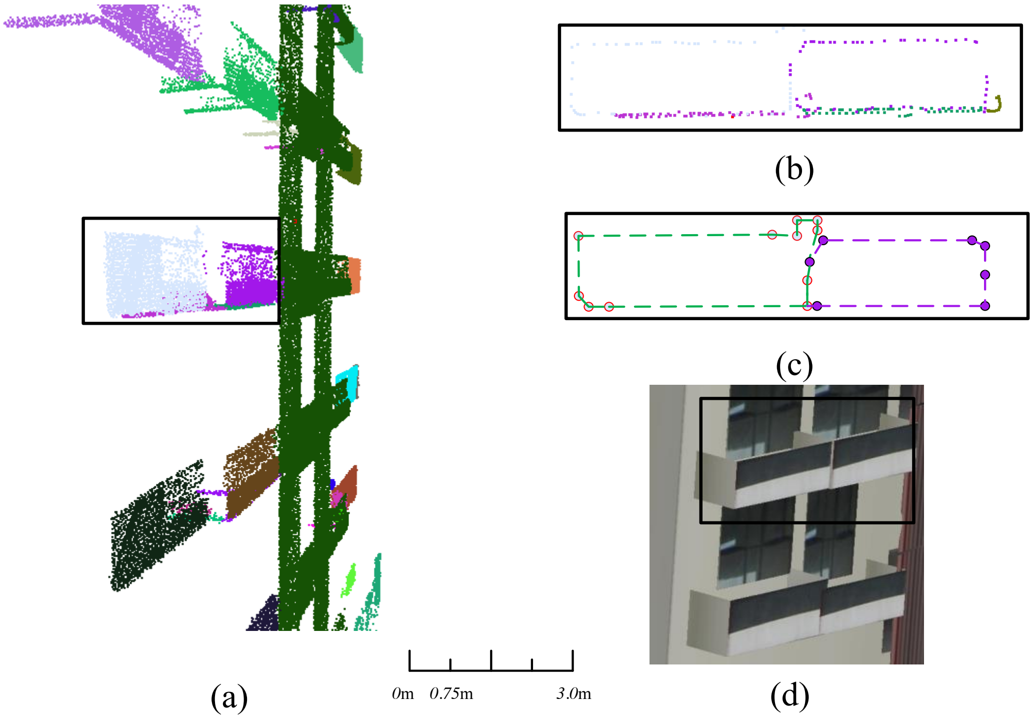


Fig. 12. Parsing a facade based on the symmetric pattern of the elements. (a) The initial segmented elements of a facade with symmetrical structures. (b) The segmented elements. (c) The initial segmented elements of a facade with incomplete symmetric patterns. (d) The segmented elements of (c).

Through the above described process, which is also illustrated by the results of Fig. 12, the position and range of each element is obtained. Then the α-shape algorithm

(Edelsbrunner *et al.*, 1983) is applied to extract the boundaries of the elements. To generate light-weight 3D facade models, we need to remove the redundant points of the element boundaries (Zhang *et al.*, 2011), and reserve the key points. In doing so, the simplification method in Chen *et al.* (2014) is applied to smooth the boundaries. The final boundary is generated by connecting the key points. At the end of the process, the 3D geometrical facade models are reconstructed by applying the algorithm of Chen *et al.* (2014).

The obtained results following the above procedure are illustrated in Fig. 13, where the depth layer decomposition and element boundary simplification can be found. The boundary points of the elements that were obtained using the α-shape algorithm (Edelsbrunner *et al.*, 1983) is illustrated in Fig. 13(b). The simplified boundaries using the method reported in Chen *et al.* (2014) are shown in Fig. 13(c). These results have been obtained by assuming that when the threshold distance is less than 0.1 m, the simplification process stops. Note that the small circles in Fig. 13(c) represent the key points of an element boundary. Fig. 13(d) shows part of the reconstructed 3D facade model that is accurate.



(a) (b) (c) (d)

Fig. 13. Depth layer decomposition and element boundary simplification. Different colors represent different planes.

6. Experimental Performance Evaluations and Discussions

To evaluate the performance of the proposed methodology, we performed both qualitative and quantitative evaluations of various TLS urban building facade point clouds with different complexities and styles. Moreover, performance comparisons with other state-of-the-art methods, such as Teboul *et al.* (2013), Martinovic *et al.* (2015), Wan and Sharf (2012) and Li *et al.* (2011), were undertaken to demonstrate the effectiveness and advantages of the proposed methodology over these already known methods. First, we present an overview of the various datasets that were used for our experiments in Section 6.1. Then, the classification results of the facade elements obtained by our method are presented in Section 6.2. Detailed performance comparisons between our method and the other previously published methods are presented qualitatively in Section 6.3 and quantitatively in Section 6.4.

6.1. Overview of the Datasets

Two types of datasets were used in our experiments: datasets readily available from the Internet and datasets that were experimentally obtained by scanners and photographs.

For the first category, the Cathedral facade point cloud of 1.45M points was downloaded from http://www-sop.inria.fr/members/Florent.Lafarge/benchmark/surface_reconstruction/reconstruction.html. Additionally, another two datasets were used for comparison purposes: the Stdh facade with 109k points (see Fig. 19) and the Apartment facade with 463k points (see Fig. 20), which were also used in (Li *et al.*, 2011).

The second category of dataset was captured by single scans of various types of building facade point clouds such as symmetric facades and highly non-symmetric facades. Shops are located on the ground floor of the Teaching and Huaqing facade buildings, whereas the other floors are residential apartments. During the scanning, the lower floors of these buildings were occluded by front trees to enable acquisition of TLS with missing data. These point clouds were acquired by RIEGL LMS-Z620 scanners (see Table 1 for scanner technical information data) using a single scan while photos of these scenes were taken with a Nikon D300 camera.

Table 1. Technical Information Data for the RIEGL LMS-Z620 Scanner.

Measurement Principle	Measurement Range	Accuracy	Precision	Beam Divergence	Laser Wave-length	Horizontal and Vertical Angle Spacing
Time of Flight	2-2000 m	10 mm	5 mm	0.15 mrad	1,550 nm	0.57°

6.2. Facade Classification

The facade classification results have an important influence on the final parsing performance. In this subsection, we present the facade classification results obtained by using the BieS-ScSPM.

In our experiment, the training and generalization sample had 15,000 pixels. The size of the regions used for extracting SIFT features was 16×16 pixels. In each sample, the number of the extracted regions ranged from 480 to 520 pixels. In the ScSPM extraction, the dimension of the sparse coding feature after max pooling was 21,054, and the sparsity parameter was λ=0.05.

To test that the BieS-ScSPM method can obtain the discriminative features of the facades, we compared the facade classification precisions using the SVM classifier based on ScSPM and BieS-ScSPM. Table 2 lists the classification precisions of the depth images based on the ScSPM and BieS-ScSPM methods. Note that the depth image had a higher generalization precision when generated by the BieS-ScSPM method.

Table 2. Precision comparison of the ScSPM and our method.

Data	ScSPM	BieS-ScSPM
Cathedral facade	40.91%	95.45%
Teaching facade	84.62%	96.15%
Huaqing facade	91.49%	97.87%
Stdh facade	85.30%	94.12%
Apartment facade	84.36%	90.63%

Fig. 14 presents the results obtained for the Huaqing facade elements. In particular, Fig. 14(a) shows the labeling results obtained by using the ScSPM algorithm, whereas Fig. 14(b) illustrates the classification results obtained by using the super-pixel. The elements bounded by the red dashed lines are the incorrectly recognized classes.

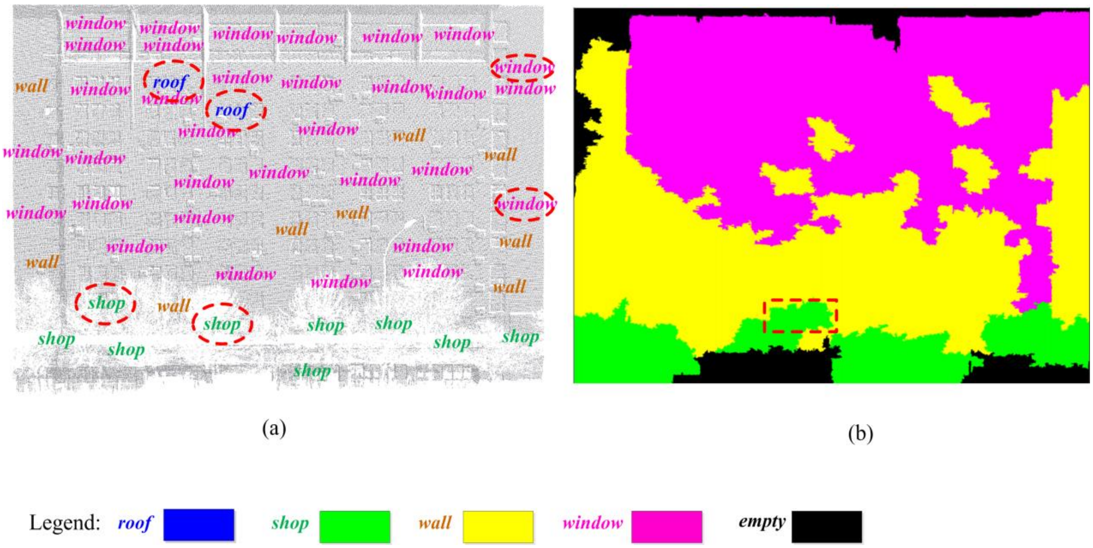

window window window window window window window window window window
wall window roof window window window window window window
window roof window wall
window window wall
window window window window
window window window wall window
window wall window wall
wall window window window
window window window wall
shop wall shop shop shop shop shop
shop shop shop
(a)
(b)
Legend: roof shop wall window empty

Fig. 14. The classification results for the Huaqing facade point cloud. (a) Elements labeled using the ScSPM. (b) The classification results based on the super-pixels. The labels in the red dashed rectangle represent the incorrect classification regions.

Fig. 15 shows the facade elements of the point cloud of the Cathedral facade that were parsed using the BieS-ScSPM. In this figure, the regions with blue boundaries are roofs, those with rose pink boundaries are windows, those with red boundaries are doors, and those with green boundaries are shops. It can also be observed that the elements and their boundaries are segmented.

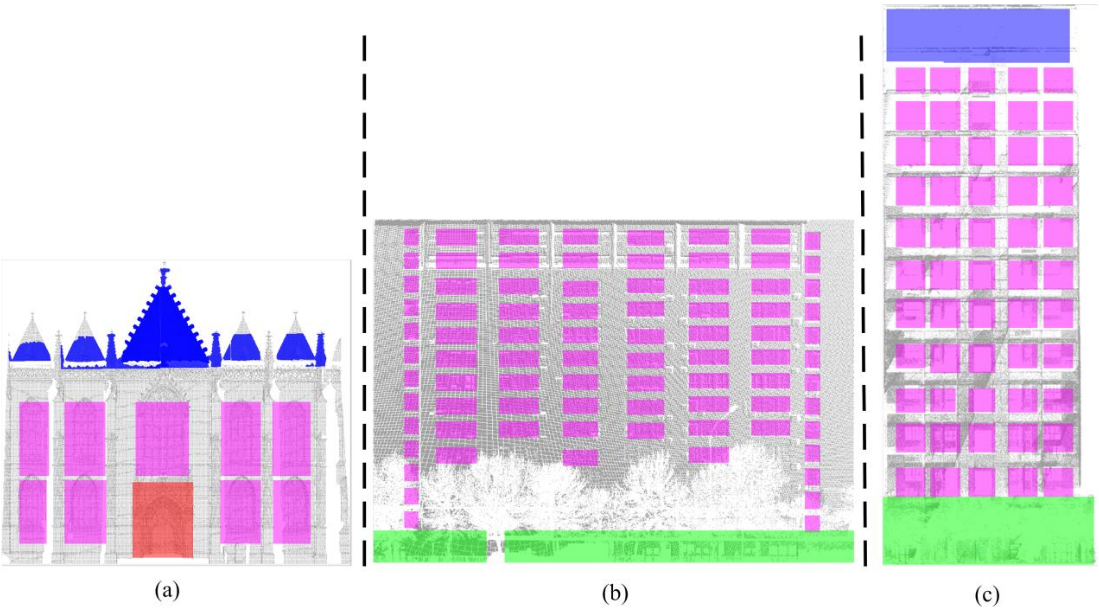


Fig. 15. Boundary extraction and optimization of the elements based on the BDE. (a) Element boundaries of the facade in the Cathedral facade point cloud. (b) Element boundaries of the Huaqing facade with missing data. (c) Element boundaries of the Teaching facade with multi-layers. The blue boundaries denote roofs, the rose pink boundaries denote windows, and the green boundaries denote shops.

6.3. Comparisons with Other Methods

In this subsection, we compare the performance of our method with that of other similar state-of-the-art methods reported in Teboul *et al*. (2013) (Method I), Wan and Sharf (2012) (Method II), Li *et al*. (2011) (Method III) and Martinovic *et al*. (2015) (Method IV) in terms of the quality of the 3D facade models.

i) Comparison with Method I

In Method I, a binary split shape grammar was presented to describe facade layouts with symmetrical and repeated structures. The parsing algorithm used was based on reinforcement learning principles and was applied to formulate the parsing problem. However, for applications involving complex facades, especially facades with irregular repetitive patterns, it is difficult to obtain good parsing performance using this method. This fact has been verified by several experimental results which will be presented next. As observed from the results shown in Fig. 16, although the roof could be segmented by Method I, it could only be parsed into a bounding box (see Fig. 16b). Notably, the parsing results are often poor if the features are not discriminative as a result of the overlap of features among different classes. Therefore, the users need to define the splitting rules, which of course makes the whole process rather inefficient and very often leads to many time consuming interactions. Fig. 17 illustrates the 3D modeling results of the Teaching facade through using the two methods. A zoom-in in Fig.17(e) shows a significant difference of the building balconies modeled by Method I from the real ones, whereas our method succeeded in obtaining a very good parsing and thus accurate modeling results (see Fig. 17d).

Fig. 18 presents the parsing results of the Huaqing facade using Method I and our method. As shown in Fig. 18(c), the shops bounded by the blue dashed rectangle are located on the ground floor, whereas the regions are split into shops under the roof. Moreover, compared with the results obtained with our method (see Fig. 18(d)), the air conditioners on the wall were not parsed in Fig. 18(c).

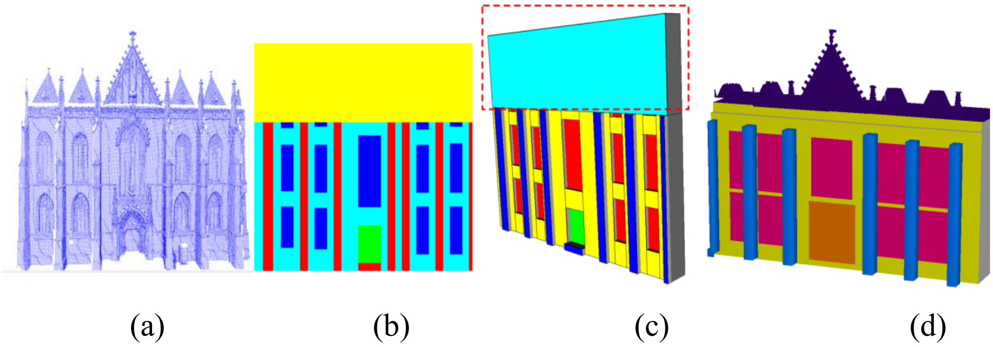


Fig. 16. Comparison of the parsing results obtained by using Method I and our

method. (a) Cathedral facade point cloud. (b) The parsing results obtained by using Method I. (c) 3D reconstruction of (b). (d) 3D reconstruction by using our method.

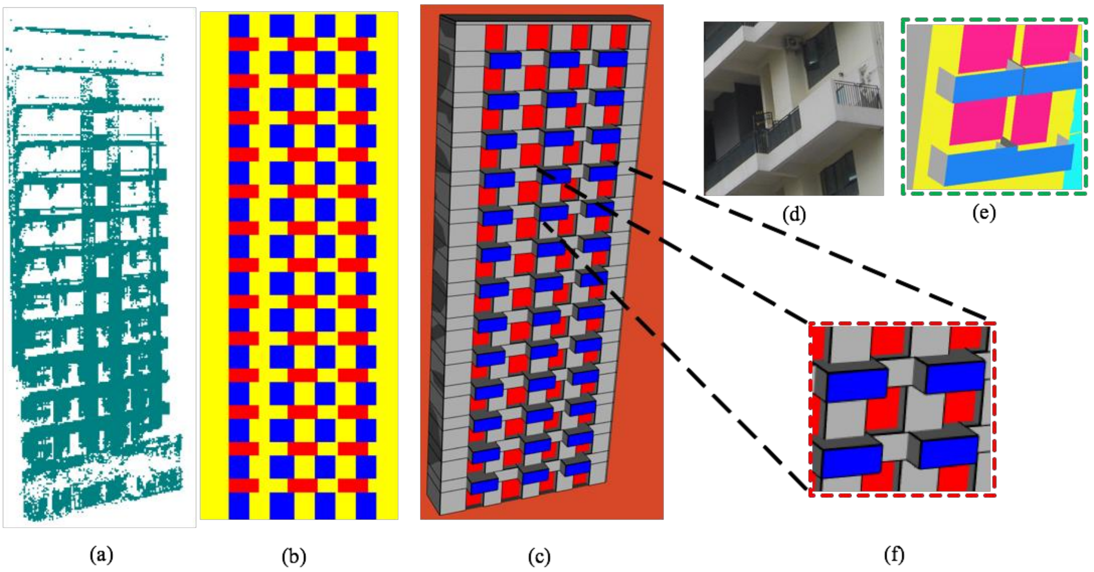


Fig. 17. 3D models of the Teaching facade reconstructed by Method I and our method. (a) Original facade point cloud. (b) The parsing result obtained by Method I. (c) 3D model reconstructed from (b). (d) The image of real balconies. (e) Zoom-in of a part of the 3D facade model obtained by our method. (f) Zoom-in of the corresponding part of the 3D model in (c).

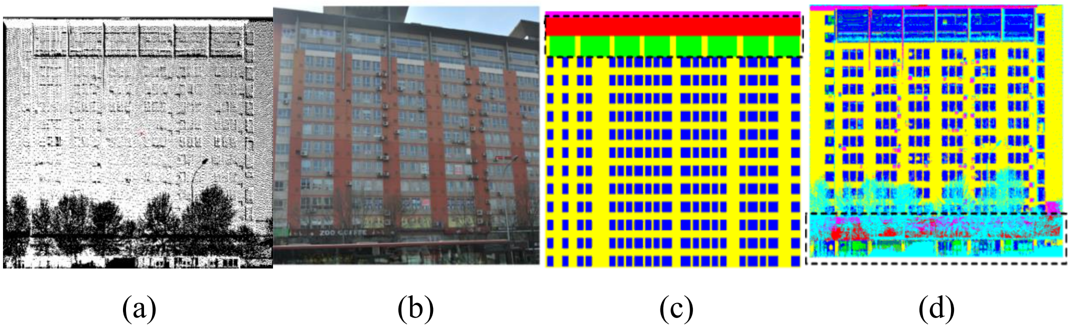


Fig. 18. Facade parsing results of Method I and our method. (a) Input Huaqing facade point cloud. (b) Photo of the facade. (c) The parsing result obtained by using Method I. (d) The parsing result obtained by using our method.

ii) Comparison with Method II

To compare the facade parsing performance of our method with that of Method II, we decided to parse a more complex building point cloud. Fig. 19 illustrates the modeling results for the Stdh facade with a highly non-symmetric pattern obtained by our method and Method II. The layer decomposition obtained by our method is shown in Fig. 19(a), whereas Figs. 19(b) and (c) show the 3D model reconstructed using Method II and our method, respectively. From the top zoom-ins of Fig. 19(b) and (c), it can be very clearly observed that our method generated more accurate boundaries of the balconies. From the middle zoom-ins of the same figures, it is also clear that our method could split the windows of the facade, whereas Method II failed to segment them. The bottom zoom-ins in these two figures show the modeling outcome of the elements with missing data. Our method can use the non-occluded structures of the facade elements to recover the boundaries of the missing regions, whereas Method II only segments facades with highly self-similarities of local geometric structures. With our method, if the incomplete data hints that the occluded structure is structured similarly to the visible elements, the occluded part is truthfully reconstructed. However, if the occluded area is different from the visible elements, then our method usually does not obtain accurate results.

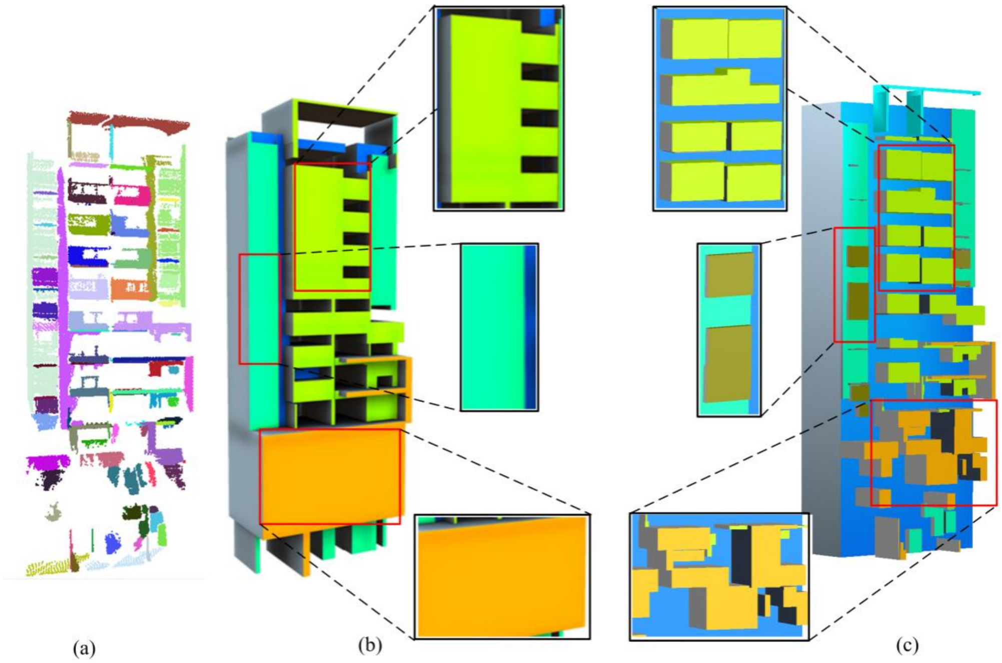


Fig. 19. 3D modeling of Stdh facade with irregular repetitive patterns. (a) Layer decomposition created by using our method. (b) The facade 3D model reconstructed by using Method II. (c) The 3D model reconstructed by using our method. The graphs in the black rectangles are the zoom-ins of the structures in the red rectangles.

iii) Comparison with Method III

To compare our method with Method III, we used the facade point cloud and the photograph that were also used in Method III and the results are shown in Fig. 20(a). Clearly this scan is of low resolution and is noisy. Fig. 20(b) illustrates the 3D facade model reconstructed using Method III. It can be observed that most of the elements were reconstructed from the noisy and sparse point cloud. However, the windows on the middle and two sides of the facade are not recognizable; e.g., those marked with a purple rectangle. On the contrary, in the 3D facade model reconstructed by our method illustrated in Fig. 20(c), the missing windows in (b) were reconstructed (illustrated in pink) that overall the derived model has more visual details and is more accurate.

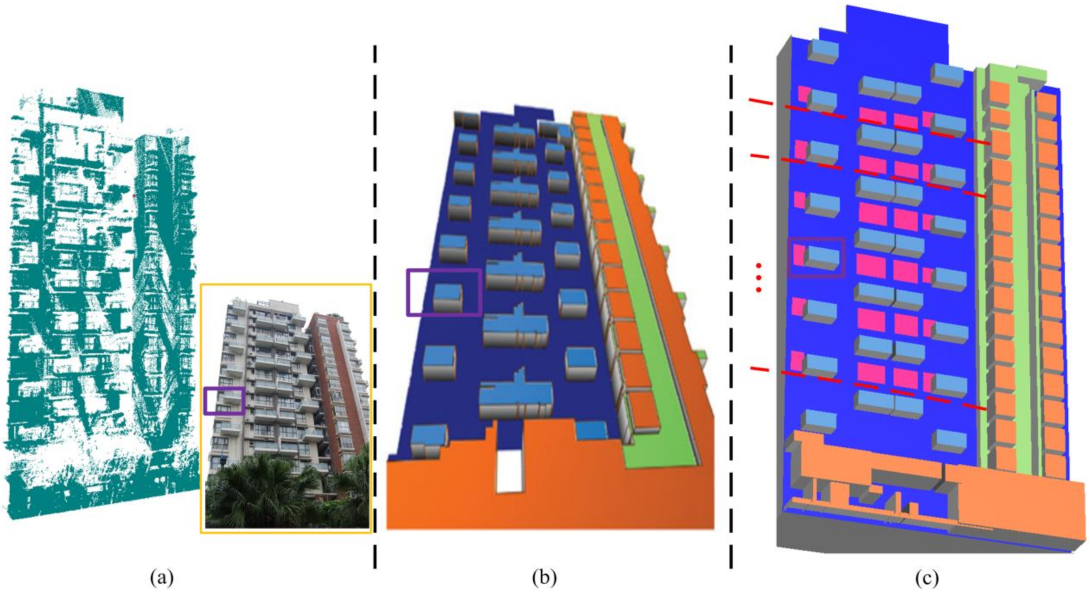


Fig. 20. Reconstruction of the Apartment facade from the incomplete and noisy point cloud. (a) The original facade point cloud (left) and photograph (right). (b) The 3D facade model reconstructed by using Method III. (c) The reconstructed facade model obtained by our method.

iv) Comparison with Method IV

Finally, we compared our method with Method IV using the input point cloud of Huaqing facade with a multi-layer pattern. Fig. 21 illustrates the reconstruction results. In Fig. 21(b), the 3D model was reconstructed by applying Method IV. This method often caused the misclassification of windows and balconies, and generated an interleaved ‘window-balcony’ pattern. Moreover, the balcony class was not recognized, mainly because air conditioners on the wall affected the relationships of the windows with the balconies. In Fig. 21(c), note that our method can reflect the depth planes of the facade and that the blue protrusions can be recognized as air conditioners or elongated column pillars. In our reconstructed 3D facade model, the air conditioners and windows have been recognized and separated because the air conditioners are adjacent to, not under, the windows, and they have a smaller size compared with the latter in the horizontal direction.

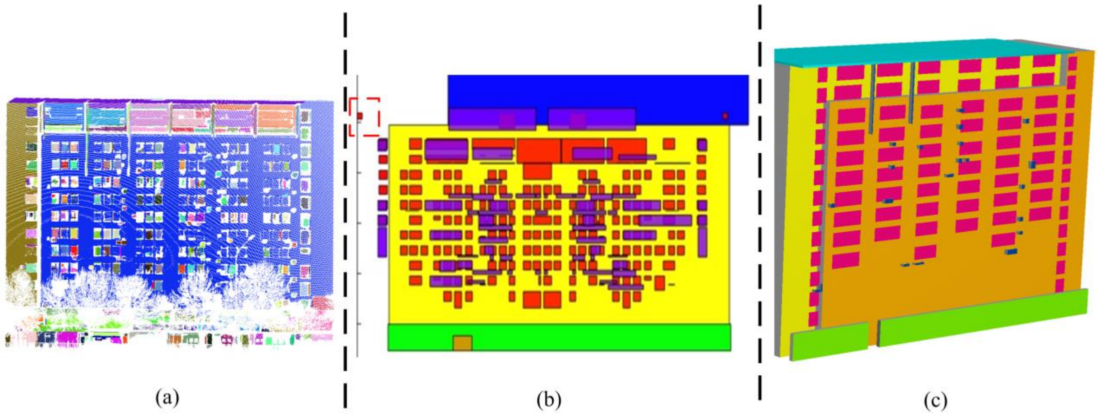


Fig. 21. Huaqing facade reconstruction. (a) Layered decomposition by our method. (b) The 3D facade model reconstructed by using Method IV. (c) The 3D facade model reconstructed by our method.

6.4 Quantitative Evaluation

In this subsection, a thorough quantitative performance comparison is presented. Our focus will be on comparing the 3D facade models reconstructed by using our method and that of the other four methods (I-IV).

Let us consider the manually segmented reference sets as the ground truth of the facade parsing results as illustrated in Fig. 22, where different colors represent different classes. For example, yellow color walls and orange color walls represent the walls which are on different layers. Then based on the ground truth, by employing our method and Methods I-IV, we have compared the number of the segmented elements and the obtained reconstruction errors summarized in Tables 3 and 4, respectively.

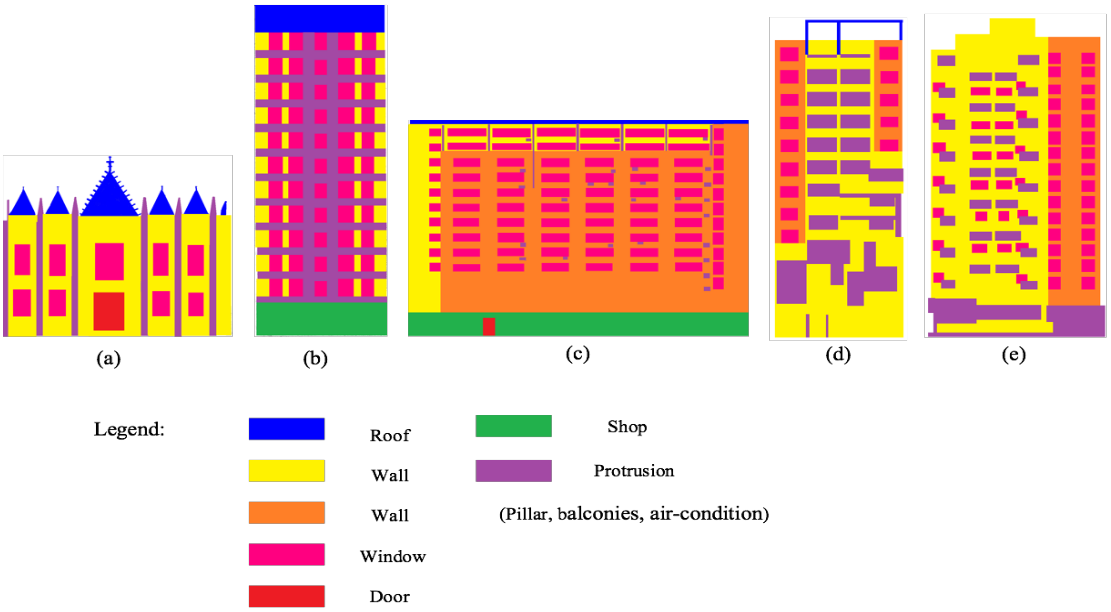


Fig. 22. The ground truths of different types of facade data. (a) Cathedral facade. (b) Teaching facade. (c) Huaqing facade. (d) Stdh facade. (e) Apartment facade.

Table 3 lists the number of the segmented elements from the facade point clouds for our method and the methods discussed in Section 6.3. It can be observed that our method segmented the elements of the facades better than the other methods.

Table 3. Number of the segmented elements from the input point clouds.

Cathedral facade	Roof	Window	Wall	Door	Protrusion	Shop	Completeness
Method I	1	13	1	1	6	-	0.920
Method II	4	12	1	0	4	-	0.733
Method III	2	10	1	1	4	-	0.822
Method IV	6	16	1	1	8	-	1.222
Our method	5	9	1	1	6	-	***0.967***
Ground truth	6	9	1	1	6	-	1.00
Teaching facade	Roof	Window	Wall	Door	Protrusion	Shop	Completeness
Method I	0	64	1	-	39	0	0.633
Method II	0	60	2	-	36	4	1.581
Method III	2	62	2	-	34	2	1.580
Method IV	2	48	1	-	41	0	0.961
Our method	1	50	1	-	40	1	***0.963***
Ground truth	1	55	1	-	44	1	1.00

Huaqing facade	Roof	Window	Wall	Door	Protrusion	Shop	Completeness
Method I	1	252	1	0	0	8	2.537
Method II	0	88	1	0	16	6	1.353
Method III	1	88	1	1	24	4	1.397
Method IV	1	168	1	1	51	1	***1.045***
Our method	1	83	2	1	21	1	0.954
Ground truth	1	81	2	1	30	1	1.00
Stdh facade	Roof	Window	Wall	Door	Protrusion	Shop	Completeness
Method I	1	18	1	-	14	-	0.821
Method II	1	0	3	-	7	-	0.583
Method III	0	15	1	-	20	-	0.589
Method IV	2	22	1	-	36	-	1.405
Our method	1	5	3	-	25	-	***0.887***
Ground truth	1	14	3	-	21	-	1.00
Apartment facade	Roof	Window	Wall	Door	Protrusion	Shop	Completeness
Method I	-	42	1	-	32	-	0.742
Method II	-	12	2	-	29	-	0.639
Method III	-	15	3	-	30	-	0.780
Method IV		45	1	-	20	-	0.617
Our method	-	54	3	-	35	-	***1.071***
Ground truth	-	56	3	-	28	-	1.00

F: the facade identifier

Table 4 compares the reconstruction error of our method with other methods. We can see from this table that the elements on the facade models reconstructed by using our method had smaller deviations from the input point datasets, which means that our models had a higher accuracy. In Table 4, we also see that Method I introduced semantic errors in parsing Huaqing facade because it failed to detect the irregular elements.

Table 4. Quantitative evaluation of the reconstruction errors.

Dataset	Methods Tested	Max (m)	Min (m)	σ (m)
Cathedral facade	Method I	1.52	0.03	0.35
	Method II	0.62	0.01	0.13
	Method III	0.40	0.01	0.05
	Method IV	1.34	0.21	0.17
	Our method	0.43	0.01	0.09
Teaching facade	Method I	0.88	0.02	0.12
	Method II	1.12	0.01	0.45
	Method III	0.48	0.06	0.07
	Method IV	0.56	0.10	0.08
	Our method	0.54	0.02	0.07
Huaqing facade	Method I	Semantic error	-	-
	Method II	0.82	0.06	0.13
	Method III	0.45	0.01	0.02
	Method IV	1.22	0.02	0.11
	Our method	0.10	0.02	0.03
Stdh facade	Method I	0.27	0.01	0.09
	Method II	0.48	0.04	0.06
	Method III	1.13	0.02	0.11
	Method IV	0.62	0.05	0.02
	Our method	0.45	0.45	0.07
Apartment facade	Method I	0.88	0.64	0.02
	Method II	1.10	0.05	0.12
	Method III	0.67	0.14	0.11
	Method IV	1.21	0.02	0.21
	Our method	0.12	0.03	0.09

Min, *Max*, and *σ* represent the minimum, maximum, and standard deviations of the distance from the TLS points to their corresponding elements (in meters), respectively.

6.5. Discussions

Both quantitative and qualitative performance evaluation results have demonstrated that the proposed approach outperforms other previously published state-of-the-art methods for facade parsing. However, it is still a challenging task for ensuring architectural plausibility when the proposed method handles point clouds with large missing data or facades with complex shapes. One possible solution way is to exploit weak architectural principles for introducing the high-level knowledge of the facades, or fuse other data sources like images or GIS data.

7. Conclusions

In this paper, we reported the development of a novel hierarchical framework for parsing and modeling building facades. Data-driven depth plane decomposition was used to segment unorganized 3D point clouds into multi-layer 2.5D depth planes. Semantic segmentation and underlying facade structures, such as repetitive and symmetric patterns, were used as priors for labeling facade elements. A high-level understanding of facade structures that allows the recognition of facade shapes and patterns was explored, and the element boundaries were then extracted for modeling 3D facades. Our method provides a complete description of the urban facades through the hierarchical framework, and it is robust with respect to noise and varying point densities. Experiments conducted on a set of TLS point clouds of various complexities and styles demonstrated that 3D facade models reconstructed using our method are more specific than those reconstructed by other related methods.

As for future work, we plan to develop an even more robust method to facilitate the full automatic detection of urban building structures from point clouds, even from those suffering from severe missing data, noise and outliers. The ultimate goal is to reconstruct high-quality urban polygonal models using LiDAR data in conjunction with other easily accessible data sources.

Acknowledgements

The authors would like to thank the reviewers and the associate editor for their thoughtful and detailed comments which have helped them to improve the scientific contribution as well as the presentation of this paper.

References

Brenner, C., 2005. Building reconstruction from images and laser scanning. International Journal of Applied Earth Observation and Geoinformation, 6(3), pp. 187-198.

Chen, D., Zhang, L., Mathiopoulos, P.T. and Huang, X., 2014. A methodology for automated segmentation and reconstruction of urban 3-D buildings from ALS point clouds. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(10), pp. 4199-4217.

Huang, H., Brenner, C. and Sester, M., 2013. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing, 79, pp. 29-43.

Edelsbrunner, H., Kirkpatrick, D. and Seidel, R., 1983. On the shape of a set of points in the plane. IEEE Transactions on Information Theory, 29(4), pp. 551-559.

Friedman, S. and Stamos, I., 2013. Online detection of repeated structures in point clouds of urban scenes for compression and registration. International Journal of Computer Vision, 102(1-3), pp. 112-128.

Hohmann, B., Havemann, S., Krispel, U. and Fellner, D., 2010. A GML shape grammar for semantically enriched 3D building models. Computers & Graphics, 34(4), pp. 322-334.

Hohmann, B., Krispel, U., Havemann, S. and Fellner, D., 2009. CityFit-High-quality urban reconstructions by fitting shape grammars to images and derived textured point clouds. In Proceedings of the 3rd ISPRS Workshop.

Huang, H., Brenner, C. and Sester, M., 2013. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing, 79, pp. 29-43.

Jagannathan, A. and Miller, E.L., 2007. Three-dimensional surface mesh segmentation using curvedness-based region growing approach. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(12), pp. 2195-2204.

Kemec, S., Duzgun, S., Zlatanova, S., Dilmen, D.I. Yalciner, A.C. 2010. Selecting 3-D urban visualization models for disaster management: Fethiye tsunami inundation case.in Proc. 3rd Int. Conf. Cartogr. GIS, Nessebar, Bulgaria. pp. 1–9.

Koziński, M., Gadde, R., Zagoruyko, S., Obozinski, G. and Marlet, R., 2015. A MRF shape prior for facade parsing with occlusions. In 2015 IEEE Conference on

Computer Vision and Pattern Recognition, pp. 2820-2828.

Lafarge, F., Descombes, X., Zerubia, J. and Pierrot-Deseilligny, M., 2008. Automatic building extraction from DEMs using an object approach and application to the 3D-city modeling. ISPRS Journal of Photogrammetry and Remote Sensing, 63(3), pp. 365-381.

Liu, M.Y., Tuzel, O., Ramalingam, S. and Chellappa, R., 2011. Entropy rate superpixel segmentation. In 2011 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2097-2104.

Li, Y., Zheng, Q., Sharf, A., Cohen-Or, D., Chen, B. and Mitra, N.J., 2011. 2D-3D fusion for layer decomposition of urban facades. In 2011 International Conference on Computer Vision, pp. 882-889.

Martinovic, A., Knopp, J., Riemenschneider, H. and Van Gool, L., 2015. 3D all the way: Semantic segmentation of urban scenes from start to end in 3D. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4456-4465.

Martinovic, A., Mathias, M., Weissenberg, J. and Van Gool, L., 2012. A three-layered approach to facade parsing. In 2012 European Conference on Computer Vision, Springer Berlin Heidelberg, pp. 416-429.

Martinovic, A. and Van Gool, L., 2013. Bayesian grammar learning for inverse procedural modeling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 201-208.

Müller, P., Zeng, G., Wonka, P. and Van Gool, L., 2007. Image-based procedural modeling of facades. ACM Transactions on Graphics, 26(3), p. 85.

Musialski, P., Wimmer, M. and Wonka, P., 2012. Interactive coherence-based facade modeling. Computer Graphics Forum, 31(2), pp. 661-670.

Nan, L., Sharf, A., Zhang, H., Cohen-Or, D. and Chen, B., 2010. SmartBoxes for interactive urban reconstruction. ACM Transactions on Graphics, 29(4), p. 93.

Pu, S. and Vosselman, G., 2009. Knowledge based reconstruction of building models from terrestrial laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing, 64(6), pp. 575-584.

Riemenschneider, H., Krispel, U., Thaller, W., Donoser, M., Havemann, S., Fellner, D. and Bischof, H., 2012. Irregular lattices for complex shape grammar facade parsing. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1640-1647.

Ripperda, N. and Brenner, C., 2009. Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments. In Proc. of the 12th AGILE Conference on GIScience, pp. 1-12.

Sampath, A. and Shan, J., 2010. Segmentation and reconstruction of polyhedral building roofs from aerial Lidar point clouds. IEEE Transactions on Geoscience and Remote Sensing, 48(3), pp. 1554-1567.

Schnabel, R., Wahl, R. and Klein, R., 2007. Efficient RANSAC for point‐cloud shape detection. Computer Graphics Forum, 26(2), pp. 214-226.

Teboul, O., Kokkinos, I., Simon, L., Koutsourakis, P. and Paragios, N., 2011. Shape grammar parsing via reinforcement learning. In 2011 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2273-2280.

Teboul, O., Kokkinos, I., Simon, L., Koutsourakis, P. and Paragios, N., 2013. Parsing facades with shape grammars and reinforcement learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(7), pp. 1744-1756.

Teeravech, K., Nagai, M., Honda, K. and Dailey, M., 2014. Discovering repetitive patterns in facade images using a RANSAC-style algorithm. ISPRS Journal of Photogrammetry and Remote Sensing, 92, pp. 38-53.

Toshev, A., Mordohai, P. and Taskar, B., 2010. Detecting and parsing architecture at city scale from range data. In 2010 IEEE Conference on Computer Vision and Pattern Recognition, pp. 398-405.

Tyleček, R. and Šára, R., 2010. A weak structure model for regular pattern recognition applied to facade images. In Asian Conference on Computer Vision, Springer Berlin Heidelberg, pp. 450-463.

Vanegas, C.A., Aliaga, D.G. and Benes, B., 2012. Automatic extraction of Manhattan-world building masses from 3D laser range scans. IEEE transactions on Visualization and Computer Graphics, 18(10), pp. 1627-1637.

Van Gool, L., Zeng, G., Van den Borre, F. and Müller, P., 2007. Towards mass-produced building models. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., pp. 209–220.

Wan, G. and Sharf, A., 2012. Grammar-based 3D facade segmentation and reconstruction. Computers & Graphics, 36(4), pp. 216-223.

Xiao, J., Fang, T., Zhao, P., Lhuillier, M. and Quan, L., 2009. Image-based street-side city modeling. ACM Transactions on Graphics, 28(5), p. 114.

Yang, J., Yu, K., Gong, Y. and Huang, T., 2009. Linear spatial pyramid matching using sparse coding for image classification. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1794-1801.

You, R.J. and Lin, B.C., 2011. A quality prediction method for building model reconstruction using LiDAR data and topographic maps. IEEE Transactions on Geoscience and Remote Sensing, 49(9), pp. 3471-3480.

Zhang, H., Xu, K., Jiang, W., Lin, J., Cohen-Or, D. and Chen, B., 2013. Layered analysis of irregular facades via symmetry maximization. ACM Transactions onGraphics, 32(4), pp. 121.

Zhang, L., Zhang, L., Tao, D., Huang, X., 2013. Tensor discriminative locality alignment for hyperspectral image spectral-spatial feature extraction. IEEE Transactions on Geoscience and Remote Sensing, 51(1), pp. 242-256.

Zhang, L., Zhang, L., Ren, Y. and Guo, Z., 2011. Transmission and visualization of large geographical maps. ISPRS Journal of Photogrammetry and Remote Sensing, 66(1), pp. 73-80.

Zhang, M., Zhang, L., Mathiopoulos, P.T., Ding, Y. and Wang, H., 2013. Perception-based shape retrieval for 3D building models. ISPRS Journal of Photogrammetry and Remote Sensing, 75, pp. 76-91.

Zhang, L., Wu, C., Du, B., 2014. Automatic radiometric normalization for multi-temporal remote sensing imagery with iterative slow feature analysis. IEEE Transactions on Geoscience and Remote Sensing, 52(10), pp. 6141-6155.

Zheng, Q., Sharf, A., Wan, G., Li, Y., Mitra, N.J., Cohen-Or, D. and Chen, B., 2010. Non-local scan consolidation for 3D urban scenes. ACM Transactions on Graphics, 29(4), p. 94.

Zhu, X.X. and Shahzad, M., 2014. Facade reconstruction using Multiview spaceborne TomoSAR point clouds. IEEE Transactions on Geoscience and Remote Sensing, 52(6), pp. 3541-3552.

Zhou, Q.Y., and Neumann, U., 2011. 5D building modeling with topology control. In 2011 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, pp. 2489–2496.