The 9th International Symposium on Mobile Mapping Technology
MMT2015, 9-11 December 2015, Sydney, Australia

A ROBUST SYSTEM OF BUILDING FACADES EXTRACTION FROM MOBILE LIDAR DATA AT STREET LEVEL

Li Li, Jian Yao, Yong Ding, Wuhan University
Kai Sun, Leador Spatial Technology Corporation

KEY WORDS: Planar Patch Detection, Building Facades Extraction, Mobile Terrestrial LiDAR

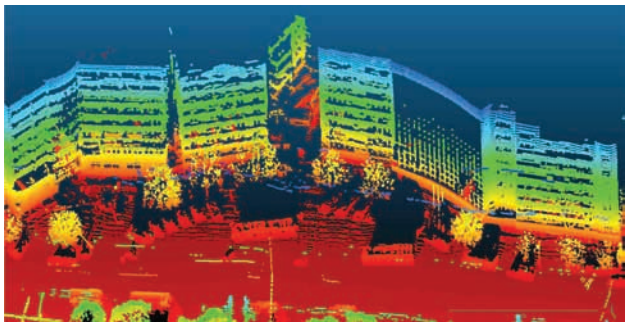
ABSTRACT:

In this paper, we present a new building facades extraction system from mobile LiDAR point clouds at street level. We firstly thin the point clouds to reduce the size of point clouds, and then the points of buildings are extracted to detect the building facades based on planar patch detection. In our method, we first partition the points into a set of voxels and fit a planar patch to all points of each voxel. Voxels that are not successfully fitted as planar patches, are iteratively partitioned into sub-voxels until reaching a minimum size. Second, we iteratively merge the planar patches and identify unclassified points based on voxel growing. The wrong planes are filtered at last. The point clouds captured from the real outdoor urban environments by Mobile Mapping System (MMS) were used to test our developed system. Experimental results in large mobile LiDAR data show that our proposed system is capable of extracting high-quality building facades and meeting the users' requirements. In addition, to prove the robustness and stability of our system, we tested it on the point clouds captured from three different kinds of laser scanners and found that it can always generate pleasant results with fixed parameter settings.

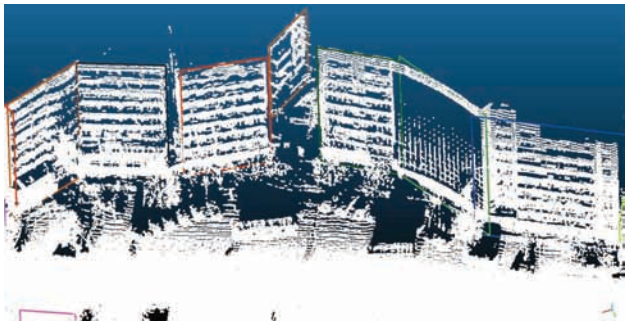
1. INTRODUCTION

Nowadays, as the continuous increasing need for detailed 3-D building descriptions in urban areas, both building extraction and 3-D reconstruction have been hot topics in the field of computer vision and photogrammetry. There are many applications of 3-D building reconstruction such as city planning, training and simulation for urban terrorism scenarios and virtual reality. Generally, the 2-D outlines of buildings must be extracted before the building descriptions can be constructed. Many research work deal with this problem using aerial or satellite images [Sahar et al., 2010, Ahmadi et al., 2010, Ok et al., 2013, Nex et al., 2013, Ok, 2013], and airborne LiDAR data [Matikainen et al., 2003, Zhang et al., 2006, Tournaire et al., 2010, Yang et al., 2013b, Xiao et al., 2014, Yan et al., 2015]. However, these approaches have the disadvantage that they can only capture the roofs of the buildings but not the facades, and their resolution is lower than mobile terrestrial laser scanned point clouds. This essential disadvantage limits their use in building facades extraction and reconstruction. The latest development in Mobile Mapping Technology (MMS) has shown that these systems are extremely efficient to acquire the panoramic images and LiDAR data at street level. Unlike airborne LiDAR data, mobile terrestrial LiDAR data can provide much more detailed information about building facades, which makes it more suitable for the extraction of building facades. Therefore, we apply the LiDAR point clouds acquired by MMS to extract building facades, a visual illustration is shown in Fig. 1. However, the problem of huge data volumes, redundancy, and GPS outages and errors affect the efficiency and accuracy of mobile terrestrial LiDAR data processing, still need to be solved.

Recently, many efforts have been made to extract building facades from mobile LiDAR data [Frueh et al., 2005, Wolf et al., 2005, Boulaassal et al., 2009, Hammoudi et al., 2009, Yang et al., 2013a]. Boulaassal et al. [Boulaassal et al., 2009] proposed two successive algorithms for automatically segmenting building facades scanned by the Terrestrial Laser Scanner. It first applied the Random Sample Consensus (RANSAC) [Fischler and Bolles, 1981] algorithm to extract planar clusters. After that, the contour



(a) LiDAR data acquired by the MMS.



(b) Outlined planar clusters of buildings. Different color rectangles represent different building facades and the white represents the points.

Figure 1: Illustration of the input/output of our proposed approach when applied on a given LiDAR data shown in (a).

points are extracted to characterize the boundary of each plane in order to be used afterwards in the 3D modelling. Hommoudi et al. [Hammoudi et al., 2009] present an approach for automatic extraction of building footprints and corresponding dominant facade planes using the terrestrial laser scanning from a MMS at street level. The proposed approach extracts and reconstructs the dominant facade planes by employing the K-mean clustering in the Hough space. Then, the building footprints are extracted by intersecting 2-D facade lines. In [Hammoudi et al., 2010, Hammoudi et al., 2011], Hommoudi et al. also employed an adapt-

ed Progressive Probabilistic Hough Transform (PPHT) to detect the dominant hypothetical facade planes . Yang et al. [Yang et al., 2013a] also developed an approach for automated footprint extraction of building facades from mobile LiDAR point clouds. It first extracted contour areas which contain points of building, points of trees, and points of other objects based on the georeferenced feature image generated from initial points. After that, the principal component analysis (PCA) method was employed to identify the points of buildings. Then, all the points in a building object were segmented into different planes using the RANSAC algorithm. Finally, footprints of different facades were generated, refined, harmonized and joined.

Since majority of building facades are planes, the key stage of our proposed system is the planar patch detection. In the past few decades, a number of plane detection techniques have been proposed. They can be roughly classified into three categories: Hough-based [Borrmann et al., 2011, Hulik et al., 2014, Camurri et al., 2014], RANSAC-based methods [Schnabel et al., 2007, Yang and Förstner, 2010, Fujiwara et al., 2013], region growing techniques [Deschaud and Goulette, 2010, Georgiev et al., 2011, Wang et al., 2013]. Hough-based method is a traditional approach to detection planes in point clouds using transformations into feature space. It utilizes a buffer in which votes for candidate planes are accumulated. An excellent example is presented in [Hulik et al., 2014]. It presents various optimization of the 3D Hough Transform used for plane detection.

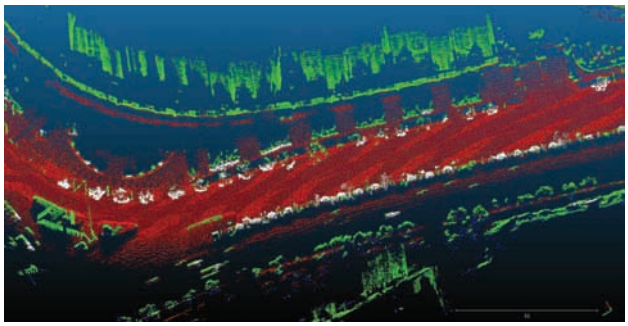
One of the most widely used methodologies for plane detection is the RANSAC algorithm. It is a randomized procedure that iteratively fits an accurate model to input data. Although the data contains a high proportion of outliers, the fitted model is also reliable. For example, Schnabel et al. [Schnabel et al., 2007] used RANSAC to detect basic shapes, such as planes, spheres, cylinders, cones and tori, in unorganized point clouds.

Region growing techniques start with a seed region and grow it by neighborhood when the neighbors satisfy some special conditions, e.g., the distance from point to plane. In range images, the spatial adjacency relations are known, but in case of unorganized 3D data, there is no information about the neighborhood in the data structure. Deschaud and Goulette [Deschaud and Goulette, 2010] proposed a fast and accurate algorithm to detect planes in unorganized point clouds using filtered normals and voxel growing.

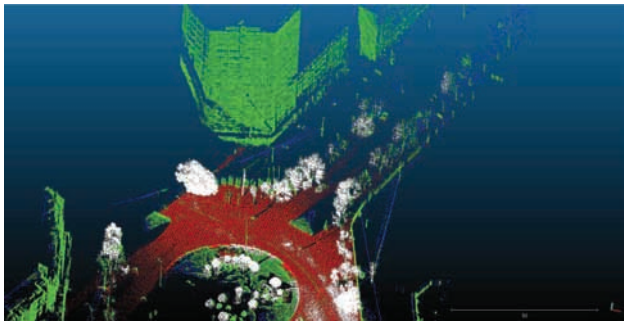
This paper presents a robust system which automatically and effectively extracts building facades from mobile terrestrial LiDAR data in complex urban environment at the street level. The proposed system involves point cloud thinning, point cloud filtering and segmentation, and building facades extraction. The point cloud thinning is used to reduce the size of data, the memory and computational time of our system. Before extracting building facades, the buildings are first filtered and segmented from point clouds, and then only detect planes in those filtered points. In addition, we proposed a novel plane detection algorithm by combining region growing and the RANSAC algorithm. Experimental results in large data collected by different kinds of laser scanners proved that our system can robustly extract high-quality building facades.

2. OUR APPROACH

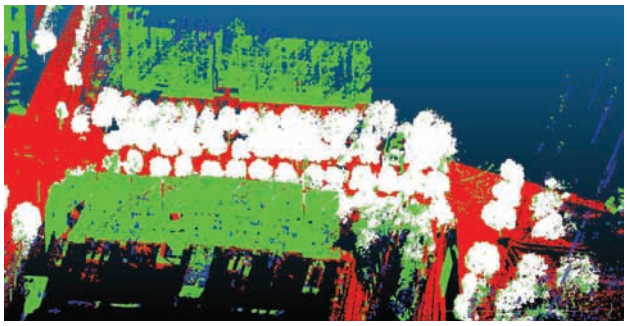
In this paper, we present a robust approach for extracting building facades from Mobile Terrestrial LiDAR points at street level.



(a)



(b)



(c)

Figure 2: Segmentation results of LiDAR point clouds acquired by (a) SICK LMS511, (b) Riegl VUX-1 and (c) Velodyne VLP-16. The green, white and red points represent the buildings, trees and floor, respectively. The blue points represent the unclassified objects, such as power lines, noise, and so on.

Our approach is comprised of three stages: preprocessing, planar patch detection and postprocessing. The main aims of preprocesssing are to reduce the size of points by thinning and segment the points of buildings. Since majority of building facades are planes, an efficient planar patch detection algorithm is proposed to extract building facades. In the stage of postprocessing, we filter the wrong planes and merge planes which belongs to the same buildings into a complete building facade. The whole steps of our proposed building facades detection approach is presented in Algorithm 1. The key stage of our approach is the proposed planar patch detection which is inspired by [Yao et al., 2010, Yao et al., 2011]. Yao et al. [Yao et al., 2010, Yao et al., 2011] proposed a plane segmentation method following a hybrid approach for range images which the spatial adjacency relations are known. This approach combines region growing and clustering strategies. It first generate a set of initial regions representing planar patches from coarse to fine, and then gradually merges and extends them into large and reliable planar patches at the region level. Finally, it iteratively refines the segmentation at the planar patches' boundaries considering their neighbors. However, in this paper,

the LiDAR point clouds which we need to deal with is different with the structured range images, are in the form of unstructured data.

Algorithm 1 Building Facades Detection

Input: The mobile terrestrial LiDAR point clouds $\mathbb{R}^3$.
Output: The building facades $\{\mathbf{F}_i\}$.

1: Points thinning: reduce the size of $\mathbb{R}^3$.
2: Points classification: obtain the points of buildings.
3: Hierarchical planar patches detection: detect rough planar patches based on hierarchical segmentation.
4: Merging planes: merge neighboring planar patches if their angle and distance are smaller than predefined thresholds.
5: Voxel growing: identify neighboring voxels for each planar patch based on voxel growing.
6: Merging planes again.
7: Filtering wrong planar patches and obtaining the last building facades $\{\mathbf{F}_i\}$.

2.1 Preprocessing

After we load the LiDAR point clouds acquired by MMS, we first reduce the size of this data by uniform sampling. Generally, the number of points after uniform sampling is less than one tenth of the number of the original input points. The memory and the computation time can be sharply reduced.

To improve the accuracy and the efficiency of planar patch detection in the next stage. We need to segment the points of building from the point clouds. We first segment the points of floor based on the value of Z coordinate based on sliding window. Then, for each point, we find their neighboring points and organize them based on k-d tree. The normal, density and curvature of every point are calculated. Generally, the density of noise and power line is small. The normal of tree is irregular and the curvature is small. On the contrary, the normal of building is regular and the curvature is large. So according to those attributes, we can distinguish those objects. Figure 2 shows several segmentation results of point classification from three kinds of typical laser scanned data.

2.2 Planar Patch Detection

In this section, we present a new algorithm to detect planar patches from unorganized 3D points acquired by MMS. The strategy we applied in planar patch detection is similar with the approach presented in [Yao et al., 2010, Yao et al., 2011]. We first obtain initial rough planar patches based on hierarchical segmentation, then the neighboring planar patches are merged if they are similar enough. Finally, we grow those planar patches by adding all neighboring points. A visual example of planar patch detection is shown in Figure 3. Due to the lack of spatial adjacency relations, we use the voxel grid as our data structure. Voxels are small cubes of the length l ($l = 1m$ was used in this paper) that partition the point cloud space. Every point belongs to a voxel and a voxel contains a set of points. Once the voxel grid has been build, we start our planar patch detection algorithm.

2.2.1 Hierarchical Planar Patches Detection Given all points $\mathbb{R}^3$, to perform a fast and robust segmentation of the points into a set of rough planar patches $\mathcal{P}$, we iteratively partition $\mathbb{R}^3$ into a set of voxels and fit their points to planar patches, which are then merged with those in their neighborhood. We first partition $\mathbb{R}^3$ into a set of voxels $\{\mathrm{B}_i\}$ of size $M \times M (M = 2^N)$. Least squares method is a popular and efficient method for fitting a plane to a set of 3D points. For each candidate voxel B_i, the least squares method is applied to fit a plane, we represent this plane as $\mathbf{P}_i$. Then, for each point $\mathbf{x}$ in B_i, we compute the distance d of the point $\mathbf{x}$ to the plane $\mathbf{P}_i$. We sort all points in an ascending order with respect to the point-to-plane distance, and find the maximal distance $d_{\max}$. If the maximal distance $d_{\max}$ is small than the predefined threshold d_{th}, this voxel is regarded to be successfully fitted. If the maximal distance $d_{\max}$ is large than d_{th}, but small than $\lambda \cdot d_{th}$ ($d_{th} = 5cm$ and $\lambda = 5$ were used in this paper), we further apply the RANSAC algorithm to fit this plane again. If $d_{\max}$ is large than $\lambda \cdot d_{th}$, this voxel is not regarded to be successfully fitted as a plane. After all fitted voxels are processed, we decrease the size of unsuccessful fitted voxels by setting $M = M/2$ and repeat the above processing until the minimum size. At last, we collect all of the successfully fitted voxels $\{\mathrm{B}_s\}$.

2.2.2 Merging Planar Patches First we collect all pairs of neighboring planar patches $\langle \mathbf{P}_p, \mathbf{P}_q \rangle$ where we assume that $\mathbf{P}_q$ contains more points than $\mathbf{P}_p$, whose corresponding voxels are B_p and B_q, respectively. Then we calculate the angle $n_{dif}(\mathbf{n}_p, \mathbf{n}_q)$ between two normals as $n_{dif}(\mathbf{n}_p, \mathbf{n}_q) = \arccos(\mathbf{n}_p, \mathbf{n}_q)$, where $\mathbf{n}_p$ and $\mathbf{n}_q$ denote the normals of $\mathbf{P}_p$ and $\mathbf{P}_q$, respectively. We also compute the mean distance $\bar{d}(\mathrm{B}_p, \mathbf{P}_q)$ of points in $\mathbf{P}_p$ to $\mathbf{P}_q$ as:

$$\bar{d}(\mathrm{B}_p, \mathbf{P}_q) = \frac{1}{|\mathrm{B}_p|} \sum_{\mathbf{x} \in \mathrm{B}_p} |d(\mathbf{x}, \mathbf{P}_q)|, \tag{1}$$

where $|\mathrm{B}_p|$ denotes the size of points in the voxel B_p, and $d(\mathbf{x}, \mathbf{P}_q)$ denotes the distance from the point $\mathbf{x}$ to the plane $\mathbf{P}_q$.

All above collected pairs are evaluated for merging as follows. If either $\mathbf{P}_p$ or $\mathbf{P}_q$ was already merged into other planar patches, we neglected this pair $\langle \mathbf{P}_p, \mathbf{P}_q \rangle$. Here $\langle \mathbf{P}_p, \mathbf{P}_q \rangle$ is merged into a single patch $\mathbf{P}'_q$ if the following conditions are satisfied:

$$\begin{cases} n_{dif}(\mathbf{n}_p, \mathbf{n}_q) < n_{th}, \\ \bar{d}(\mathrm{B}_p, \mathbf{P}_q) < \bar{d}_{th}, \end{cases} \tag{2}$$

where n_{th} and $\bar{d}_{th}$ denote the angle threshold and the mean distance threshold, respectively, and they were set as $n_{th} = 10^\circ$, $\bar{d}_{th} = 0.1m$ in this paper.

We iteratively collect pairs of neighboring planar patches and merge them until there are no new merged planar patches.

2.2.3 Voxel Growing We collect all merged planar patches $\{\mathbf{P}_m\}$ generated in Section 2.2.2, and regard them as seed planes, whose corresponding voxels are $\{\mathrm{B}_m\}$. Instead of growing with k nearest neighbors, we grow with voxels. In this paper, we test points in 26 voxel neighbors and first sort these planar patches into an descending order according to the number of voxels.

All above planar patches are orderly grown as follow. For each plane $\mathbf{P}_m$, we compute the distance to the current plane for each point in the neighbor voxel. Let N be the number of points whose distances are smaller than a predefined threshold v'_{th}. If $N \leq \gamma |\mathrm{B}_m|$ ($\gamma = 0.4$ was used in this paper), we add this voxel into B_m. We iteratively add voxels into B_m until there are no new added voxels. Then, we applied the RANSAC algorithm to fit this plane again and obtain the new plane $\mathbf{P}'_m$. Finally, we iteratively increase the value of v'_{th} until the maximum threshold v_{th} ($v_{th} = 0.6m$ was used in this paper), and repeat the above growing processing.

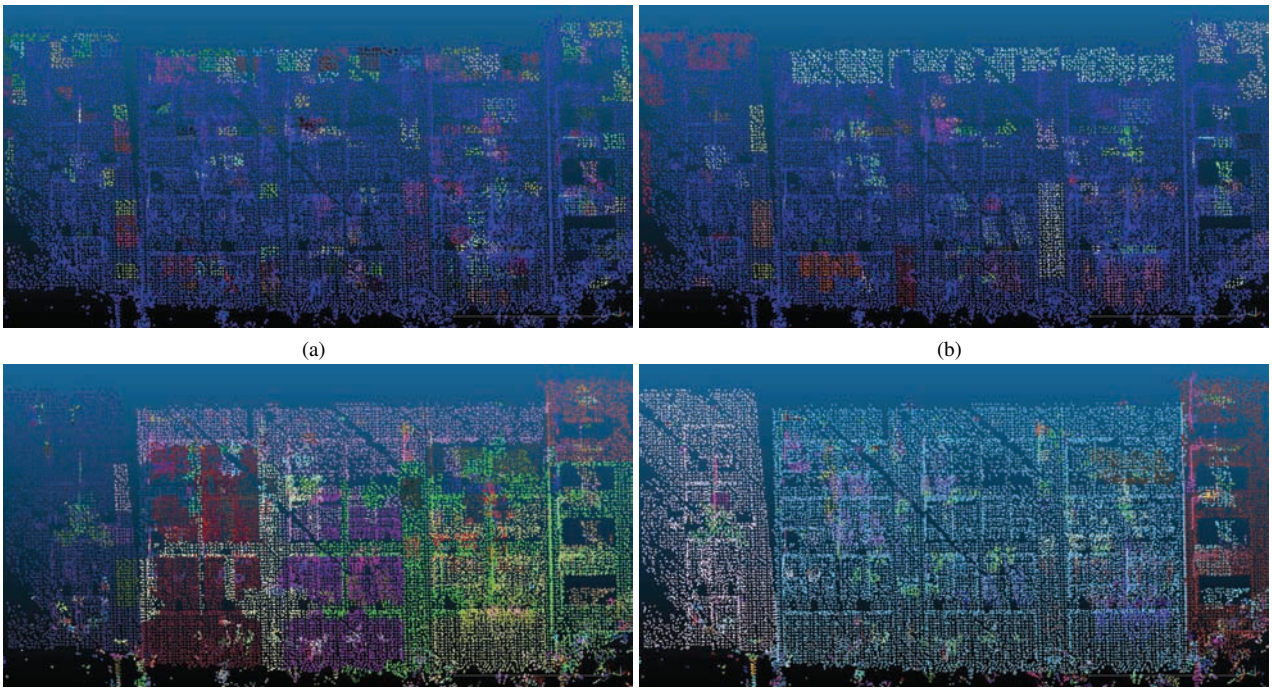


Figure 3: Illustration of our proposed planar patch detection approach: (a) Initial planar patches detected by hierarchical segmentation; (b) The merged results of (a); (c) The planar patches after voxel growing; (d) The merged results of (c). The different colors represent different fitted planar patches, except for the blue points in (a) and (b) which belong to unsuccessful fitted voxels.

At last, we merge all planar patches according to the method described in Section 2.2.2, two thresholds used in the process of plane merging are set as $n_{th} = 30^{\circ}$ and $\bar{d}_{th} = 1.5m$, and then we obtain the last planar patches $\{\mathbf{P}_g\}$, as shown in Figure 3(d).

2.3 Postprocessing

In the Section 2.2, we have detected all planar patches from input LiDAR point clouds. As we know, majority of building facades are planes, but some objects can be also segmented as a plane. So we need to filter the planar patches which don't represent the building facades. In addition, we use the rectangle model to represent the building facades at last.

As we know, the building must be vertical to the earth, so we first filter the horizontal planar patches according to the normals calculated in the process of the planar patch detection, only the planar patches with the approximate vertical normals are remained. For those vertical planar patches, we represent them with the largest bounding rectangles. The width and height of rectangles can be computed, and the area and the width-to-height ratio can be further calculated. If the area or the ratio is small, we further filter them.

According to the priori knowledges of building facades, generally, in the front of a building, there are no more building facades. So for each rectangle, we judge whether there is a more bigger rectangle behind it, if it is, we filter this rectangle, otherwise, we remain it. In this step, many rectangles which represent the cars, trees and other objects can be filter. Because they are almost located in the front of buildings.

At last, we collect all building facades represented by rectangles, and denote them as $\{\mathbf{F}_i\}$.

3. EXPERIMENTAL RESULTS

We test our proposed approach on the point clouds captured from several urban scenes by different kinds of laser scanners installed on the mobile vehicle platform. Our proposed approach was validated on, but not limited to the point clouds acquired by the SICK LMS511, Riegl VUX-1, and Velodyne VLP-16 laser scanners. In the following experiments, we call them as SICK, Riegl and Velodyne for convenience. Three SICK laser scanners were installed on the mobile vehicle platform to scan the left, right and front scenes of the moving vehicle, and we merged those point clouds into a unified whole for extracting the building facades. For Riegl and Velodyne, only one laser scanner is needed to acquired the point clouds of the street. From the vehicle trajectory provided by GPS, the length of the used data can be computed easily. All parameters of our proposed approach remain unchanged as defined in Section 2. for all experiments. Our algorithm was implemented with C++ under Windows and tested in a computer with an Intel Core i7-4770 at 3.4GHz and a 16GB memory.

Figure 4 shows the result of building facades extracted form the points acquired by SICK in the Huangshi city of China. The length of this acquired data is 33.5km and the size of those data is 11.7Gb. The computation time of our proposed approach is 2.43 hours. From the detailed local regions of Figure 4(a), we observed that the dominant facades were well-extracted. For example, in Figure 4(b), there are mainly two buildings. Our proposed approach successfully extracted them and represented them with three rectangles, and the similar results can be found in another local regions. However, the telegraph pole was also extracted as building facades which need to be filtered in the postprocessing, and we will solve this problem in the future work. In Figure 4(d), we present the overhead view of our results, which proved that our approach can be used to generate high-quality footprints of buildings. In Figure 4(g), we found that our approach can accurately extract the building facades in spite of many trees in the

front of those buildings. We also present the results of building facades in the suburban environments, as shown in Figures 4(i)(j). Generally, the buildings in the suburban region are shorter and smaller than the urban region. At the same time, the distances to the road are more longer, so the points of buildings are sparse and imprecise. But, the building facades are also well-extracted. The building facades shown in Figures 4(c)(e)(f)(h)(i) were all well-extracted by our proposed approach.

Figure 5 shows the results of building facades extracted from the point clouds acquired by Riegl in the Jingzhou city of China. The length of this data is 22.7km and the size of those data is 47.7Gb. The computation time of our proposed approach is 5.38 hours. Compared with the point clouds acquired by SICK, we observed that this data is more denser, the size of this data is more larger, so the computation time of this data is more longer. We also found that the problem of noise in this data is severe mainly due to the effective scanning distance of Riegl is too long. However, the dominant facades of buildings are also well-extracted, which fully proved that our proposed approach is robust with the noise. In Figure 5(c), we show a residential region with many similar buildings, from which we found that the points is incomplete and discontinuous in the side of those houses, but it didn't affect the completeness of the facades. In Figure 5(e), the structure of this building is complex, but our approach can also handle it. In Figure 5(g) where the main objects are buildings, trees, power lines and telegraph poles, we found that the dominant facades were detected but the planar patches of telegraph pole were also retained just like Figure 4(a). More detailed local regions of Figure 5(a) are shown in Figures 5(b)(d)(f)(h)-(j).

Figure 6 shows the results of building facades extracted from the points acquired by Velodyne in the Wuhan city of China. The length of this data is 6.68km and the size of those data is 9.66Gb. The computation time of our proposed approach is 1.61 hours. From Figures 6(b)(c)(e)-(g), we observed that all building facades were extracted completely and accurately, and there are few wrong facades in the results. However, in Figure 6(d), we also found that a tall building fail to be represented as a complete rectangle. According to our analysis, there are main two reasons. One is that this building is so tall that the points in its top is inaccurate, so the top points can't be fitted as a plane or added to the fitted planes in the process of voxel growing. Therefore, the rectangles can't cover the top of this building. Another is that the facade of this building is curved, so can't be fitted as one complete planar patch. So, we found that there are four rectangles in this building.

4. CONCLUSION

In this paper, we proposed a robust approach for extracting building facades from the mobile terrestrial LiDAR point clouds acquired from MMS at street level. To improve the efficiency of building facades extraction, we first reduce the size of point clouds and find all the points of buildings for building facades detection. A new effective planar patch detection algorithm was proposed to find building facades based on voxel growing technique and the RANSAC algorithm, which have been widely used for the planar patch detection. At last, we filter the wrong planes and collect the building facades from all detected planar patches. We tested our proposed approach on the large point clouds acquired by MMS in real urban environments with different kinds of laser scanners. Experimental results show that our proposed approach is capable of extracting high-quality building facades and meeting the requirements of 3D reconstruction.

REFERENCES

Ahmadi, S., Zoej, M. V., Ebadi, H., Moghaddam, H. A. and Mohammadzadeh, A., 2010. Automatic urban building boundary extraction from high resolution aerial images using an innovative model of active contours. International Journal of Applied Earth Observation and Geoinformation 12, pp. 150–157.

Borrmann, D., Elseberg, J., Lingemann, K. and Nüchter, A., 2011. The 3D hough transform for plane detection in point clouds: A review and a new accumulator design. 3D Research 2, pp. 1–13.

Boulaassal, Landes and Grussenmeyer, 2009. Automatic extraction of planar clusters and their contours on building façades recorded by terrestrial laser scanner. International Journal of Architectural Computing 7, pp. 1–20.

Camurri, M., Vezzani, R. and Cucchiara, R., 2014. 3D hough transform for sphere recognition on point clouds. Machine Vision and Applications 25, pp. 1877–1891.

Deschaud, J.-E. and Goulette, F., 2010. A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing. In: International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT).

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, pp. 381–395.

Frueh, C., Jain, S. and Zakhor, A., 2005. Data processing algorithms for generating textured 3D building facade meshes from laser scans and camera images. International Journal of Computer Vision 61, pp. 159–184.

Fujiwara, T., Kamegawa, T. and Gofuku, A., 2013. Plane detection to improve 3D scanning speed using RANSAC algorithm. In: IEEE Conference on Industrial Electronics and Applications (ICIEA).

Georgiev, K., Creed, R. T. and Lakaemper, R., 2011. Fast plane extraction in 3D range data based on line segments. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

Hammoudi, K., Dornaika, F. and Paparoditis, N., 2009. Extracting building footprints from 3D point clouds using terrestrial laser scanning at street level. ISPRS/CMRT09 38, pp. 65–70.

Hammoudi, K., Dornaika, F. and Paparoditis, N., 2011. Generating virtual 3d model of urban street facades by fusing terrestrial multi-source data. In: International Conference on Intelligent Environments (IE).

Hammoudi, K., Dornaika, F., Soheilian, B. and Paparoditis, N., 2010. Extracting wire-frame models of street facades from 3D point clouds and the corresponding cadastral map. Remote Sensing and Spatial Information Sciences 38, pp. 91–96.

Hulik, R., Spanel, M., Smrz, P. and Materna, Z., 2014. Continuous plane detection in point-cloud data based on 3D hough transform. Journal of Visual Communication and Image Representation 25, pp. 86–97.

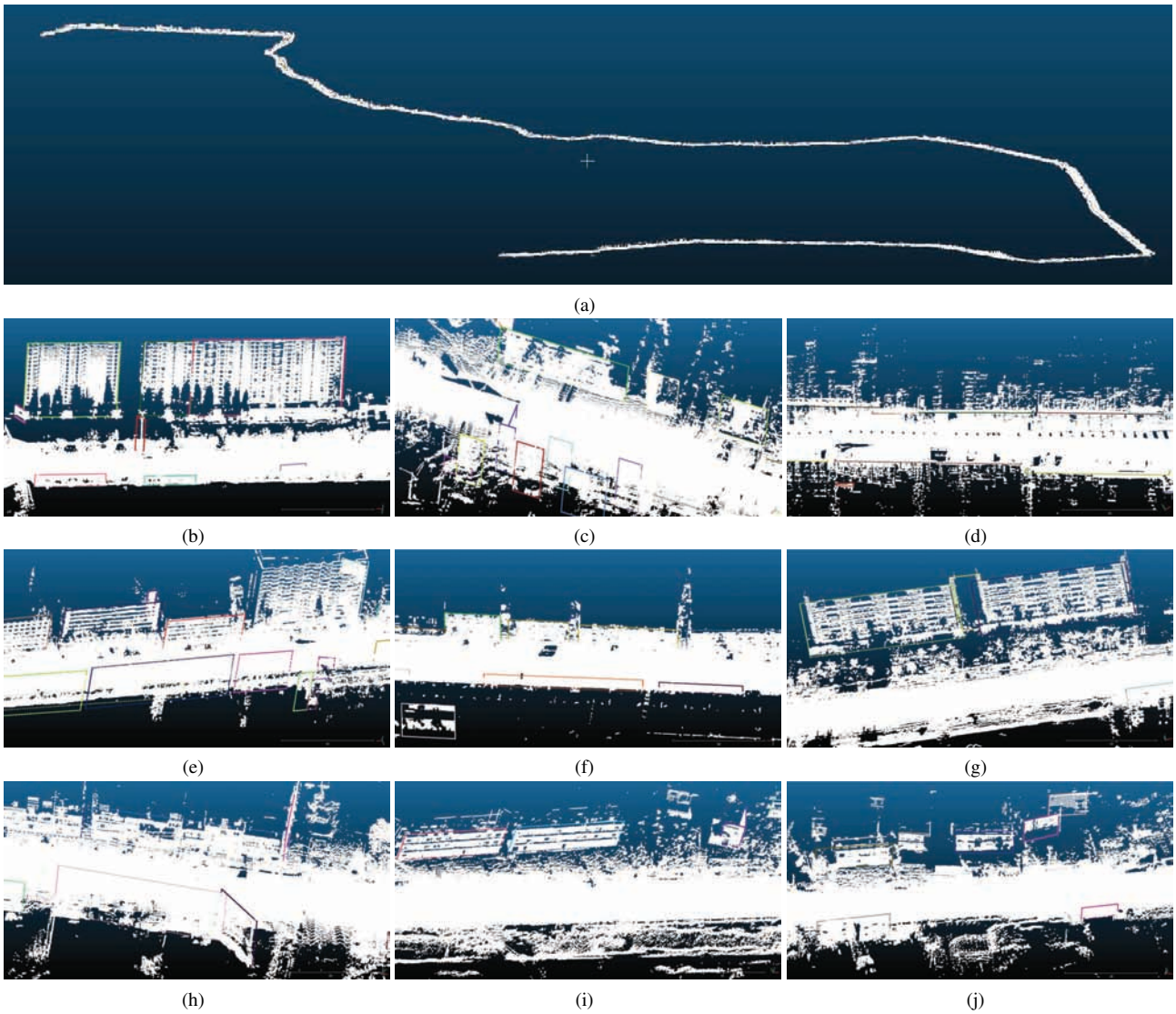


Figure 4: The building facades detection results of the SICK laser scanner: (a) The overview of building facades extraction results; (b)-(j) The local results of (a). The length of this data is 33.5km and the size of this data is 11.7GB. The computation time of our proposed approach is 2.43 hours. The rectangles with different colors represent different building facades.

Matikainen, L., Hyyppä, J. and Hyyppä, H., 2003. Automatic detection of buildings from laser scanner data for map updating. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 34, pp. 218–224.

Nex, F., Rupnik, E. and Remondino, F., 2013. Building footprints extraction from oblique imagery. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci 2, pp. 61–66.

Ok, A. O., 2013. Automated detection of buildings from single VHR multispectral images using shadow information and graph cuts. ISPRS Journal of Photogrammetry and Remote Sensing 86, pp. 21–40.

Ok, A. O., Senaras, C. and Yuksel, B., 2013. Automated detection of arbitrarily shaped buildings in complex environments from monocular VHR optical satellite imagery. IEEE Transactions on Geoscience and Remote Sensing 51, pp. 1701–1717.

Sahar, L., Muthukumar, S. and French, S. P., 2010. Using aerial imagery and GIS in automated building footprint extraction and shape recognition for earthquake risk assessment of urban inventories. IEEE Transactions on Geoscience and Remote Sensing 48, pp. 3511–3520.

Schnabel, R., Wahl, R. and Klein, R., 2007. Efficient RANSAC for point-cloud shape detection. In: Computer Graphics Forum.

Tournaire, O., Brédif, M., Boldo, D. and Durupt, M., 2010. An efficient stochastic approach for building footprint extraction from digital elevation models. ISPRS Journal of Photogrammetry and Remote Sensing 65, pp. 317–327.

Wang, T., Chen, L. and Chen, Q., 2013. A graph-based plane segmentation approach for noisy point clouds. In: Chinese Control and Decision Conference (CCDC).

Wolf, D., Howard, A. and Sukhatme, G. S., 2005. Towards geometric 3D mapping of outdoor environments using mobile robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

Xiao, Y., Wang, C., Li, J., Zhang, W., Xi, X., Wang, C. and Dong, P., 2014. Building segmentation and modeling from airborne lidar data. International Journal of Digital Earth 5, pp. 1–16.

Yan, J., Zhang, K., Zhang, C., Chen, S.-C. and Narasimhan, G., 2015. Automatic construction of 3-D building model from air-

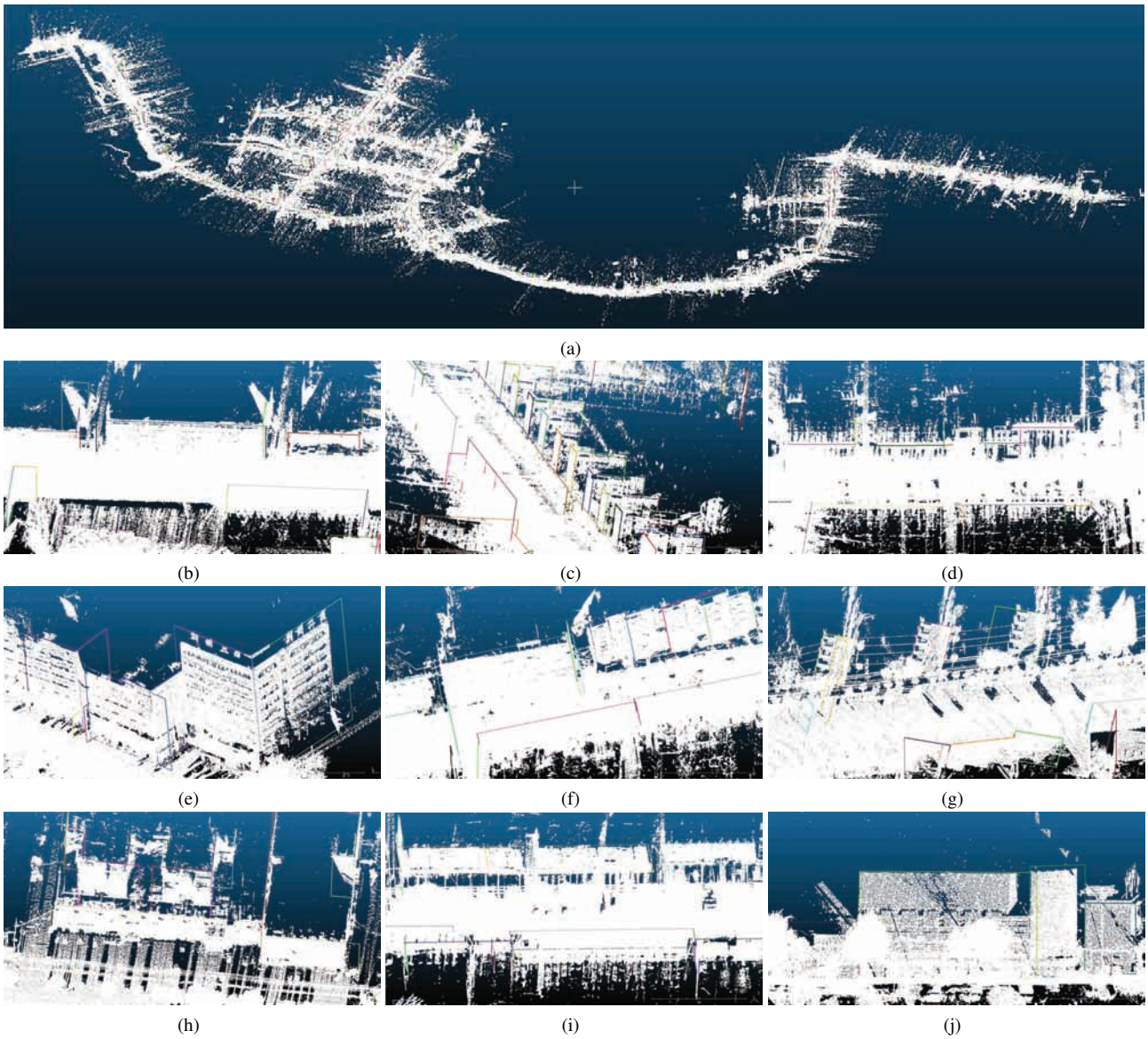


Figure 5: The building facades detection results of the Riegl laser scanner: (a) The overview of building facades extraction results; (b)-(j) The local results of (a). The length of this data is 22.7km and the size of this data is 47.7GB. The computation time of our proposed approach is 5.38 hours. The rectangles with different colors represent different building facades.

borne LIDAR data through 2-D snake algorithm. IEEE Transactions on Geoscience and Remote Sensing 53, pp. 3–14.

Yang, B., Wei, Z., Li, Q. and Li, J., 2013a. Semiautomated building facade footprint extraction from mobile LiDAR point clouds. IEEE Geoscience and Remote Sensing Letters 10, pp. 766–770.

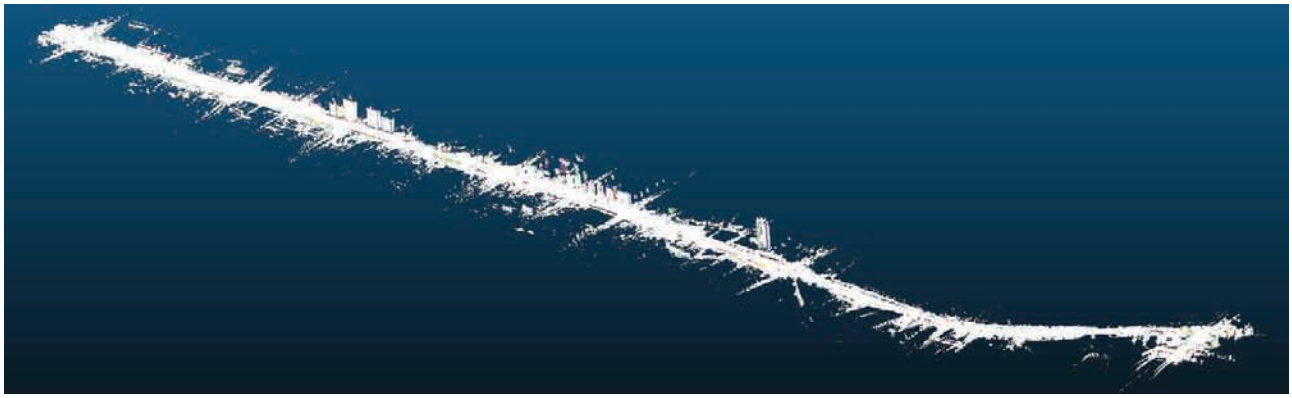
Yang, B., Xu, W. and Dong, Z., 2013b. Automated extraction of building outlines from airborne laser scanning point clouds. IEEE Geoscience and Remote Sensing Letters 10, pp. 1399–1403.

Yang, M. Y. and Förstner, W., 2010. Plane detection in point cloud data. In: International Conference on Machine Control & Guidance.

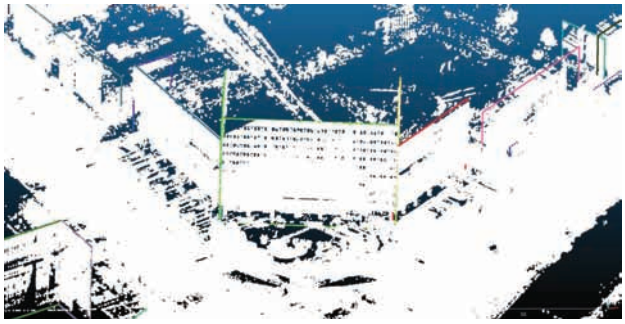
Yao, J., Ruggeri, M. R., Taddei, P. and Sequeira, V., 2010. Automatic scan registration using 3D linear and planar features. 3D Research 1, pp. 1–18.

Yao, J., Taddei, P., Ruggeri, M. R. and Sequeira, V., 2011. Complex and photo-realistic scene representation based on range planar segmentation and model fusion. The International Journal of Robotics Research 30, pp. 1263–1283.

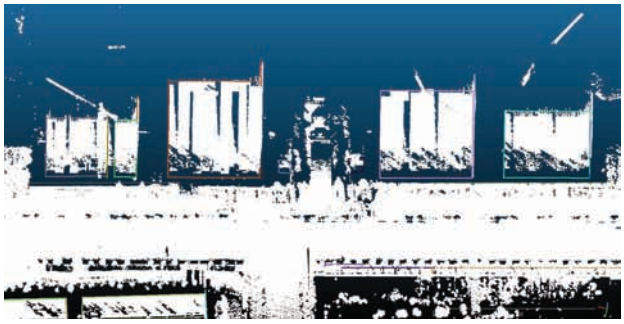
Zhang, K., Yan, J. and Chen, S.-C., 2006. Automatic construction of building footprints from airborne LIDAR data. IEEE Transactions on Geoscience and Remote Sensing 44, pp. 2523–2533.



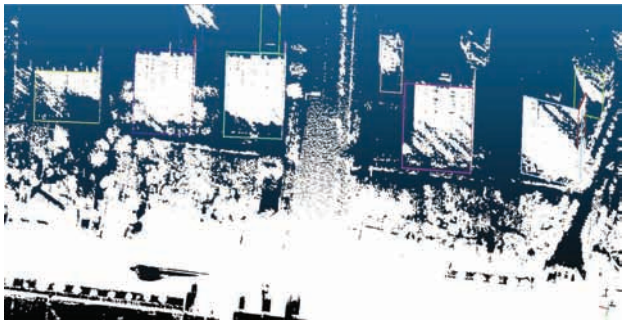
(a)



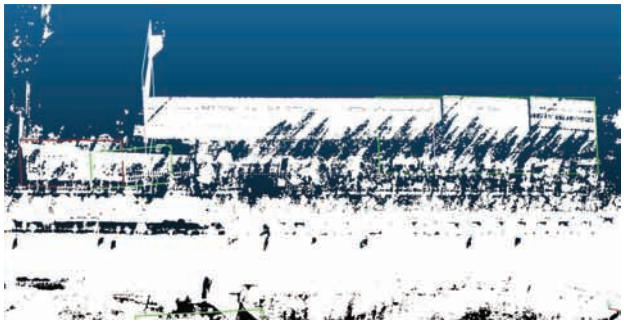
(b)



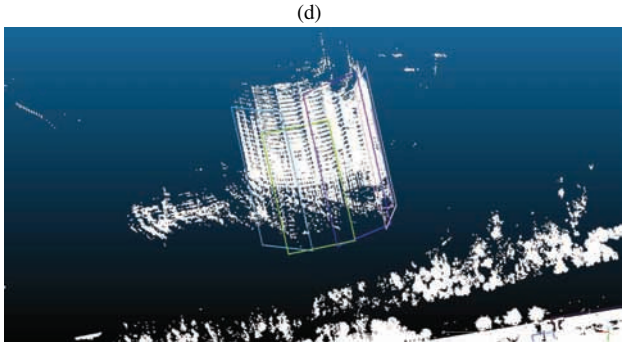
(c)



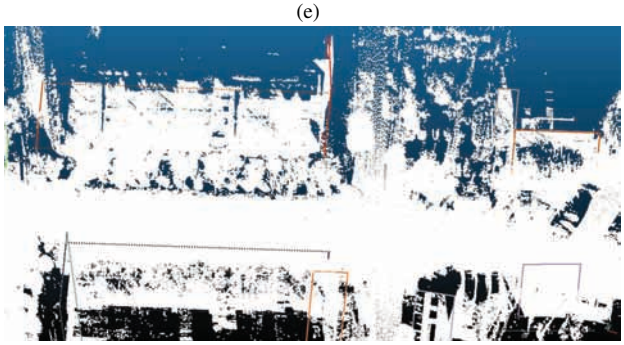
(d)



(e)



(f)



(g)

Figure 6: The building facades detection results of the Velodyne laser scanner: (a) The overview of building facades extraction results; (b)-(e) The local results of (a). The length of this data is 6.68km and the size of this data is 9.66GB. The computation time of our proposed approach is 1.61 hours. The rectangles with different colors represent different building facades.