EXTRACTION OF BUILDING WINDOWS FROM MOBILE LASER SCANNING POINT CLOUDS

Menglan Zhou[1], Lingfei Ma[1], Ying Li[1], Jonathan Li[1][2]*

[1] WatMos Lab, Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada, *Corresponding author: junli@uwaterloo.ca

[2] VIP Group, Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

ABSTRACT

This study recognizes the significance and considerable commercial applications in creating Level of Detail (LoD) building models for 3D city models generation. Accordingly, this paper proposes a novel method to identify and extract window frames on building facades from Mobile Laser Scanning (MLS) point clouds. The proposed method can typically be regarded as a stepwise procedure. Firstly, a voxel-based upward-growing method is applied to distinguish non-ground points from ground points. Next, outliers are filtered out from non-ground points by statistical analysis. Then, all the remaining non-ground points are clustered based on the conditional Euclidean clustering algorithm to segment out building facades. A volumetric box is afterward created to store façade points so that neighbors of each point can be operated. Finally, a manipulator is applied according to the structural characteristics of window frames to extract the potential window points. Quantitative evaluations based on 2D validation and 3D validation were both conducted. In the 2D validation, the lowest F1-measure of the test datasets is 0.740, and the highest can be 0.977. While in the 3D validation, the lowest precision of the test dataset is 79.58%, and the highest can be 97.96%. The results demonstrate the proposed method can successfully extract the rectangular or curved windows in the test datasets with promising accuracies to support the generation of LoD3 building models.

Index Terms **-** Building window, feature extraction, mobile laser scanning (MLS), point cloud, LoD3 building model.

I. INTRODUCTION

Nowadays, mobile LiDAR or mobile laser scanning (MLS) point clouds are applicable in cosmopolitan building model reconstruction due to its high flexibility and acquisition rate in large-scaled complex scenes [1][2]. However, the most challenging work of the LoD3 building modeling from the noisy point clouds is to extract detailed features on facades, such as windows, doors, and remove holes caused by systematic errors [4]. Windows are indispensable frames on building facades, nevertheless, there are no laser points for window glasses due to its very low reflectivity. In addition, because of aesthetic requirements and cultural diversities, windows are designed in multitudinous shapes [5].

The primary step of window extraction is conducting façade extraction from MLS point clouds. Existing methods in building facades extraction are mainly based on the generated geo-referenced images, coordinate information, shape features, and prior knowledge [1], [6]-[9]. Nevertheless, there still no sophisticated algorithms that can effectively classify 3D MLS point clouds collected in large-scale complex environments into semantic objects. The window extraction is usually the subsequent procedure of the building façade extraction. Many studies were conducted in recent years [5], [10]-[12]. However, prior semantic knowledge, including sizes of windows and intervals between windows, is essentially needed.

Accordingly, developing a reliable method to extract windows while retaining their geometry, semantic, and coordinate information precisely from the noisy MLS point clouds has become a colossal challenge in recent years. This paper will focus on establishing a supporting rationale and developing a semi-automated algorithm for 3D window extractions from MLS point clouds. This algorithm can typically be regarded as a stepwise procedure to interpret MLS point clouds as semantic features: 1) A voxel-based upward-growing method is first applied to distinguish non-ground points from ground points. 2) Noise is then filtered out from non-ground points by statistical analysis. 3) In order to segment the building facades, all the remaining non-ground points are clustered based on the conditional Euclidean clustering algorithm. 4) Clusters whose density and width are over a predefined threshold will be designated as points for building facades. 5) After a building façade is successfully extracted, a volumetric box is created to store façade points so that neighbors of each point can be operated. 6) A manipulator is finally applied to extract the potential window points based on the structural characteristics of window frames.

II. METHOD

In this section, a voxel-based upward-growing algorithm is conducted to filter out ground points from the raw MLS point clouds. After that, noise still inevitably exists in the MLS point clouds, which are primarily presented as isolated points, outliers, and point mutations in local areas. In order to remove noise from the non-ground features, a statistical analysis filter is applied to differentiate noise from non-ground features [13]. However, there are no clear topological relationships between points in a discrete and sparse 3D non-ground point clouds. In order to distinguish specific 3D objects from these discrete and unorganized non-ground points, the conditional Euclidean clustering method is utilized to conduct fast segmentation of the 3D unorganized non-ground points [2].

A. ***Building Façade Extraction by Density/Width Analysis***

In this section, a density/width analysis method is put forward based on the density and geometric properties of building facades in point clouds. The building façade extraction method by density/width analysis proposed in this section is inspired by Melzer's culling mechanism. In the culling mechanism, Melzer indicated that features in 3D point clouds had different density properties [14]; the algorithm implemented in this section is based on his assertion.

The X-Y plane is firstly subdivided into a 2D grid by g m× g m, in which g is pre-defined by average point density of input clusters generated by the conditional Euclidean clustering algorithm. The input clusters are then projected into this gridded X-Y plane with the z value of each point is remaining as a label and the total number of this cluster will be recorded as N_c. For each cluster, the length Wid and the average density Den of the projected cluster can be calculated as follows:

$$W_1 = x_{max} - x_{min} \tag{1}$$

$$W_2 = y_{max} - y_{min} \tag{2}$$

$$Wid = \begin{cases} W_1, & W_1 > W_2 \\ W_2, & W_1 < W_2 \end{cases} \tag{3}$$

$$Area = \begin{cases} W_1 \times W_2, & W_1 \neq 0 \text{ and } W_2 \neq 0 \\ W_1, & W_2 = 0 \\ W_2, & W_1 = 0 \end{cases} \tag{4}$$

$$Den = N_c / Area \tag{5}$$

where x_{max} is the maximum x value in the input cluster, y_{max} is the maximum y value in the input cluster, x_{min} is the minimum x value in the input cluster, and y_{min} is the minimum y value of the input cluster, $Area$ is the pseudo rectangular acreage of the projected area. Especially, when the $W_1 = 0$ and $W_2 = 0$, the input cluster will be directly regarded as non-façade points.

Under the prior knowledge that clusters belonging to building facades have a relatively high average density and width at the same time, clusters whose average density and width are inside a certain thresholding interval will be regarded as building facades. The remaining clusters will be processed in the next step, where the given thresholding interval is determined by densities of raw 3D point clouds.

B. ***Window Extraction by Hole Detection***

Extracting windows from laser point clouds can rely on window edge extractions on walls, then windows can be localized by using points of walls. The hole detection algorithm utilized in this section is inspired by [5], who developed an operator to extract windows based on a hole-detection algorithm. The method developed in this section refines the algorithm proposed in [5], because such method [5] acquired prior knowledge of window sizes and intervals between windows. A pattern that classified window frames into four categories is firstly conducted: horizontal borders on the top and bottom of windows as well as vertical borders on the left and right side of windows according to the characteristics that potential windows always leave holes on building facades. In order to simulate neighborhood relationships among points, a volumetric manipulator at a grid size v_g is created to contain all the points of the extracted building façade, where v_g is determined by the density of the input point cloud. Then an operator is conducted according to the window pattern to localize windows excluding window crossbars. For each voxel (i, j, k), I bespeak f (i, j, k) =0 if there is no laser point in this voxel, and f (i, j, k) =1 if there are laser points in this voxel, the equation of the operator lists below:

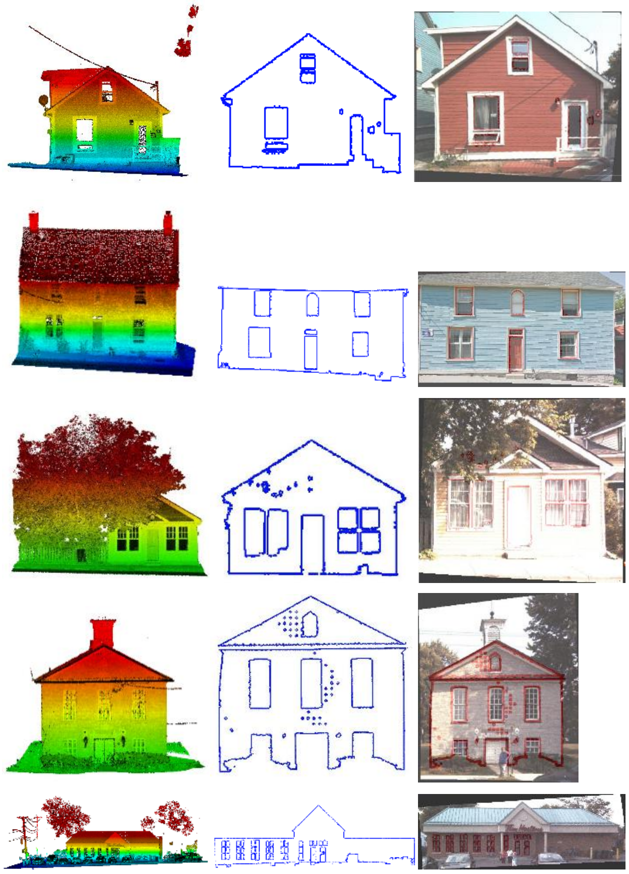
$$\begin{cases} \text{horizontal window edge points}, & \text{if } \{\sum_{k'=k}^{k'=k+1}\sum_i\sum_j f(i,j,k')\} = 1 \ \&\&\{\sum_{k'=k}^{k'=k-1}\sum_i\sum_j f(i,j,k')\} = 0 \&\&\{\sum_i\sum_j f(i,j,k)\} = 1 \\ & \text{OR if } \{\sum_{k'=k}^{k'=k-1}\sum_i\sum_j f(i,j,k')\} = 1 \ \&\&\{\sum_{k'=k}^{k'=k+1}\sum_i\sum_j f(i,j,k')\} = 0 \ \&\&\{\sum_i\sum_j f(i,j,k)\} = 1 \\ \text{vertical window edge points}, & \text{if } \{\sum_{i'=i}^{i'=i+1}\sum_j\sum_k f(i',j,k)\} = 1 \ \&\&\{\sum_{i'=i}^{i'=i-1}\sum_j\sum_k f(i',j,k)\} = 0 \ \&\&\{\sum_i\sum_j f(i,j,k)\} = 1 \\ & \text{OR if } \{\sum_{i'=i}^{i'=i-1}\sum_j\sum_k f(i',j,k)\} = 1 \ \&\&\{\sum_{i'=i}^{i'=i+1}\sum_j\sum_k f(i',j,k)\} = 0 \ \&\&\{\sum_i\sum_j f(i,j,k)\} = 1 \\ \text{non}-\text{window edge points}, & \text{otherwise} \end{cases} \tag{6}$$

As shown in Eq. (6), the upper horizontal window border is recognized if the upper neighbor exists while the lower neighbor does not; the same rationale is manipulated to the lower border, left border, and right border of windows. When window frames are successfully localized by this mechanism, points belonging to these four window borders will be designated as a classifier and marked as window points.

III. Validation

A. ***Window Regions in 2D***

Digital images are used as the references for the result validation. Digital images and 3D point clouds are collected together while the vehicle is moving so that the



(a) (b) (c)

Fig. 1. Window extraction results on five point cloud datasets used in this paper.

two kinds of sensors (optical camera and laser scanner) share the same environmental variables. However, digital images have distortions, therefore, in this study, orthophotos of building façades should be generated to validate the extracted 3D windows before the accuracy assessment. When orthophotos are successfully generated, the extracted windows will be overlapped on the orthophotos to generate the overlapped photos. The resolutions of the referenced orthophoto and the overlapped photo for each dataset are adjusted to be the same to avoid the influences caused by different resolutions.

The accuracy assessment mechanism of window regions in 2D is conducted based on recall, precision, F1-*measure*. As shown in Eqs. (7) to (9), the recall represents the completeness of the extracted windows, the precision shows how many valid and correct windows are extracted by using the proposed method, and *F1-measure* is a global score by integrating precision and recall. Where C_p represents the number of valid pixels belonging to the exact windows in the extracted points, R_p is the number of the window pixels interpreted manually from the generated orthophotos, and R_r is the number of pixels of the extracted points by the proposed method.

$$precision = C_p/R_r \tag{7}$$

$$recall = C_p/R_p \tag{8}$$

$$F1-measure = 2 \times \frac{precision \times recall}{precision + recall} \tag{9}$$

B. ***Window Regions in 3D***

The accuracy assessment of window regions in 3D is based on precision. Window points are manually extracted from the raw point clouds in this section. For each 3D point extracted by the proposed method, a corresponding point extracted by the manual interpretation should be found. The precision shows the portion of correct window points extracted by the proposed method. It is defined as C_n/R_n, where C_n is the number of valid 3D window points extracted by the proposed method those can be found in manually interpreted points, and R_n is the total number of extracted 3D window points by the proposed method.

VI. RESULTS AND DISCUSSION

A. ***Point Cloud Data***

Fig. 1 (a) shows the five point-cloud datasets that were selected from the data acquired by the RIEGL VMX-250 system. They are (from left-upper to left-bottom in Fig. 1) a single-detached house with 787,235 points in about 86.79 m^2, a single-detached house covered by 1,442,607 points in about 185.31 m^2, a single-detached house covered by 1,281,313 points in about 212.94 m^2, a single-detached house with 725,579 points in about 324.86 m^2, and a coffee shop with 3,336,229 points in about 1,577.45 m^2, respectively. Those houses include typical window types (e.g., rectangular and arc-rounded windows). Dataset 3 was used to test whether the proposed method is influenced by occlusions of trees. Dataset 5 was used to validate the reliability of the proposed method in a complex scene. In addition, holes usually exist on building facades in 3D point clouds due to systematic errors in the mobile LiDAR system. Therefore, Dataset 1 was used to test the influences that big holes will have on the hole detection algorithm.

B. ***Window Extraction Result***

The building façade extraction results obtained by the density/width analysis are influenced by the two parameters: Den_b (a predefined average density threshold of building facades) and Wid_b (a predefined width threshold of building facades). This experiment set Den_b=4000 pts/m2, and Wid_b=8 m for all the test datasets.

To guarantee the time complexity as well as the precision of this algorithm, there should be no more than 10 points in the same voxel, so the size of a voxel was set as v_g = 0.05 m in this section to ensure there cannot be more than 2 points in a voxel. The promising results indicate that the proposed window extraction algorithm can successfully extract all windows on the test datasets, including rectangular, irregular, and arc-rounded windows (see Fig. 1 (b)). In addition, as shown in the results of Dataset 3, the proposed method is not influenced by occlusions of trees. However, some big holes caused by systematic errors of the MLS system are also extracted from the raw point clouds. Furthermore, as shown in Fig. 1 (c), only window frames on the front façade in Dataset 4 can be extracted.

Table I. 2D and 3D performance evaluation

Methods / Datasets	2D performance evaluation			3D performance evaluation		
	Precision	Recall	F1-measure	C_n	R_n	Precision
1	79.67%	95.40%	0.868	3035	3814	79.58%
2	97.60%	85.17%	0.910	8638	9061	95.34%
3	95.42%	72.74%	0.826	7000	7274	96.23%
4	88.54%	63.51%	0.740	4495	5226	86.12%
5	97.79%	97.63%	0.977	7445	7600	97.96%

C. *Window Regions in 2D*

As shown in Table I, the precision, recall, and *F1-measure* of Dataset 1 are relatively low since big holes in the raw point clouds are also extracted by the hole-detection algorithm. In Datasets 2, 3 and 4, the precision is adequately high but the recall is relatively low.

As shown in Fig. 1(c), curtains are dropped down when the data was collecting. In addition, the windows are recessed on the building façade and they are not on the same vertical plane as the building façade. Therefore, only points of outer frames of the windows are put into the same cluster as the building façade. As a result, the outer frames of the windows in the three datasets can be extracted, while the inner crossbars of the windows are removed. The precision, recall, and *F1-measure* of Dataset 5 are all very high since there are no impacts of curtains, holes, or occlusions on the raw point clouds.

D. *Window Regions in 3D*

The results of 3D performance evaluation are also listed in Table I. Compared with the performance evaluation results in Table I, it can be concluded that the values of the precision in the test datasets in 2D and 3D are basically aligned with each other. The precision values of Datasets 2, 3 and 5 are 95.34%, 96.23%, and 97.96%, respectively. Such results prove that the proposed method can extract accurate 3D window frames when there are no defects in the raw point clouds. However, the precision values of Datasets 1 and 4 are 79.58%, and 86.12%, respectively, which are lower among the test datasets. It reveals that the big holes caused by system errors have considerably negative effects on the proposed method.

V. CONCLUDING REMARKS

In this paper, we have presented a method that can semi-automatically extract 3D window points from MLS point clouds to support the LoD3 building modeling. Five datasets were tested in this study to prove the feasibility of the proposed method. The *F1-measure* of the five test datasets in the 2D validation are 0.737, 0.910, 0.826, 0.740, and 0.977, respectively. The precision of the five test datasets in 3D validation are 79.58%, 95.34%, 96.23%, 86.12% and 97.96%, respectively. After a detailed analysis of the experimental results, we could conclude that the proposed method can successfully extract all types of 3D windows and glass doors (including rectangular, irregular, and arc-rounded ones) from the test datasets with promising accuracies. However, for those windows with curtains drawn, concave in walls or with holes in raw point clouds, the accuracy of the proposed method was considerably affected. In addition, the experimental results suggest that the proposed method is not affected by tree occlusions.

ACKNOWLEDGEMENTS

The work was supported by an NSERC Engage Grant. The datasets used in this study was provided by Tulloch Engineering Ltd.

REFERENCES

[1] Yang, B., Wei, Z., Li, Q., and Li, J. Semiautomated building facade footprint extraction from mobile LiDAR point clouds. *IEEE Geosc. Remote Sens. Lett.*, vol. 10, pp. 766-770, 2013.

[2] Lin, Y., Wang, C., Cheng, J., Chen, B., Jia, F., Chen, Z., and Li, J. Line segment extraction for large scale unorganized point clouds. *ISPRS J. Photogra. Remote Sens.* pp. 172-183, 2015.

[3] Yu Y., Li, J., Guan, H., and Wang, C. Automated detection of three-dimensional cars in mobile laser scanning point clouds using DBM-Hough-Forests. *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4130-4142, 2016.

[4] Aijazi, A. K., Checchin, P., and Trassoudaine, L. Automatic detection and feature estimation of windows from mobile terrestraial LiDAR data. *Intelligent Autonomous System.*, vol. 13, pp. 659-674, 2016.

[5] Wang, R., Ferrie, F. and Macfarlane, J. A method for detecting windows from mobile LiDAR data. *Photogra. Engine. Remote Sens.*, vol. 78, no. 11, pp.1129-1140, 2012.

[6] Arachchige, N. H., Perera, S. N. and Mass, H. G. Automatic processing of mobile laser scanner point clouds for building facade detection. *ISPRS Archives*, vol. 39, no. 5, pp. 187-192, 2012.

[7] Lin, Y., Wang, C., Chen, B., Zai, D., and Li, J. Facet segmentation-based line segment extraction for large-scale point clouds. *IEEE Geosc. Remote Sens.*, vol. 55, no. 9, pp. 4839-4854, 2017.

[8] Pu, S., and Vosselman, G. Extracting windows from terrestrial laser scanning. *ISPRS Archives*, vol. 36, pp. 12-14, 2007.

[9] Pu, S., Rutzinger, M., Vosselman, G. and Elberink, S. O. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS J. Photogra. Remote Sens.*, vol. 66, no. 6, pp. 28-39, 2011.

[10] Aijazi, A. K., Checchin, P., and Trassoudaine, L. Automatic detection and feature estimation of windows in 3D urban point clouds exploiting façade symmetry and temporal correspondences. *Int. J. Remote Sens.*, vol. *35*, no. 22, pp. 7726-7748, 2014.

[11] Mesolongitis, A. and Stamos, I. Detection of windows in point clouds of urban scenes. *2012 IEEE CVPR Worksh.*, pp. 17-24, 2012.

[12] Nguatem, W., Drauschke, M., and Mayer, H. Localization of windows and doors in 3D point clouds of facades. *ISPRS Annals*, vol. 2, no. 3, pp. 87-94, 2014.

[13] Rusu, R. B. and Cousins, S. 3D is here: Point cloud library (PCL). *2011 IEEE Inter. Conf. Robotics Autom.*, pp. 1-4, 2011.

[14] Melzer, T. and Briese, C. Extraction and Modeling of Power Lines from ALS Point Clouds. NA., pp. 1-8, 2004.