A Semi-Automatic Method for Building Boundary Extraction from Airborne Lidar Data

You Shao, Australia Samsung Lim, Australia

Key words: Lidar; Building Extraction; Filtering; Vegetation Detection; Semi-automatic Processing

SUMMARY

This paper presents a semi-automatic approach to building boundary extraction using airborne lidar data. Three steps are proposed and implemented directly on the lidar points rather than on the rasters obtained from the lidar data. First, ground points and non-ground points are separated using an adaptive morphological filter in order to remove the ground points. Then a fusion of methods including Normalized Difference Vegetation Index (NDVI), hierarchical clustering and thresholding is employed to further remove unwanted objects such as trees and cars. Finally, the boundary polygons are extracted and delineated based on alpha-shape and Douglas-Peucker algorithms. The extracted polygons are assessed in terms of 11 indices categorized in three evaluation methods. The test results show that the proposed method can accurately extract urban residential buildings from airborne lidar data.

A Semi-Automatic Method for Building Boundary Extraction from Airborne Lidar Data

You Shao, Australia Samsung Lim, Australia

1. INTRODUCTION

Urban areas are of increasing importance in most countries since they have been changing rapidly over time. Buildings are the main objects of interest in these areas, and building boundaries are one of the key factors for urban mapping and city modelling (Potsiou 2010). Hence, accurate building boundary extraction has been a prevalent topic of many research efforts (Forlani, Nardinocchi et al. 2006, Sohn and Dowman 2007, Dong, Kyoung et al. 2008, Huang and Zhang 2011). However, building boundary extraction from airborne lidar data is challenging due to the complexity of building shapes and the irregular distribution of lidar data points. Also, lidar data provides less accurate information about building edges compared to its vertical accuracy (Awrangjeb, Ravanbakhsh et al. 2010). Hence the extracted boundaries tend to be uneven and irregular, even if building points are accurately classified (Akkiraju, Edelsbrunner et al. 1995, Wei, Jin et al. 2011). In order to obtain more reliable boundaries, we need to regularize the extracted edge lines (Sampath and Shan 2007, Tan and Wang 2011). Although there are plenty of algorithms to resolve the aforementioned problems, any single method cannot deal with all situations at once whilst each algorithm can perform well under a certain condition and with a particular requirement. The existing approaches to building extraction using lidar data usually include three steps: ground filtering, separation of features, and extraction of building boundaries (Dash, Steinle et al. 2004, Keqi, Jianhua et al. 2006, Sohn and Dowman 2007, Awrangjeb, Ravanbakhsh et al. 2010, Awrangjeb, Zhang et al. 2013); while some researchers merged the filtering and separation steps into one as segmentation or classification (Miliaresis and Kokkas 2007). Though there are different methods to extract the building boundaries, the criteria to differentiate buildings from other features are based on the same physical characteristics, such as elevation, steepness, and homogeneity (Meng, Currit et al. 2010).

In this paper, we attempt to apply suitable methods to airborne lidar data and to assess the quality based on three evaluation methods. We propose a semi-automatic approach to building boundary extraction from lidar data. The entire approach is solely point-based processing. That is, image processing based on rasters will not be used. Our concern is to preserve the shapes as much as possible since rasterisation may result in a significant amount of loss of details due to the inappropriate choices of cell sizes and interpolation methods. The trade-off is that point-based methods are difficult to achieve acceptable running speeds. The proposed approach includes three steps: ground filtering, vegetation removal and boundary polygon delineation. First, an adaptive morphological filter is developed and performed on the raw lidar point cloud data to separate ground and non-ground points. Then points of vegetation and small non-building features are removed using a fusion of Normalised Difference Vegetation Index (NDVI), hierarchical clustering, and thresholding. Finally the boundary polygons are obtained from the classified building points using a combined boundary extraction method. Evaluation of our results is carried out based on

both visual examination and statistical analysis. A reference of building polygons digitized from an ortho-image is used in the evaluation. A *t*-test is used to compare the difference between the reference and our results and to obtain the statistics of the test results.

2. SEPARATION OF GROUND AND NON-GROUND

2.1 Filtering Methods and Inspirations

The most fundamental challenge in lidar applications is the separation of ground points and those falling onto objects. The separation approaches can be categorised as either grid-based filtering or point-based filtering (Mahmoud and Trinder 2010). Many research efforts have been made to grid-based filtering. Grid-based filters usually convert lidar points into regular grids, which is known as rasterisation, and then apply image processing algorithms. For example, Kilian et al. (1996) utilized a filtering method based on mathematical morphology, which identifies objects in a greyscale image by applying morphological operations such as opening and closing. In case of point-based filtering, some researchers used Triangular Irregular Networks (TIN) to represent the ground surface in lidar points. Axelsson (1999, 2000) developed an adaptive filtering algorithm based on TIN, which can manage surfaces with discontinuities in urban areas. However, only a few filters directly work on lidar points. Since it is unrealistic to process a huge amount of lidar points at once, it is reasonable to use small tiles or windows to scan over the whole data. Therefore, we adapted the morphological filter of Kilian et al.'s (J. Kilian 1996) for point-based processing. That is, a programme is developed to work directly on points rather than images resulted from rasterisation.

The determination of the window size is still a critical issue to this method, since a large window size results in an increased omission error by removing too many non-ground points while a small window size leads to a significant commission error by identifying non-ground points as ground ones. One way to deal with this problem is to use different window sizes and assign different weights to the results. For example, Zhang et al. (2003) proposed a progressive morphological filter that repeats the process several times with gradually increased window sizes. However, the choice of the assigned weights can still be an issue. In our study, we added an adaptive function that can automatically detect a size of the above-ground features and then change window size accordingly. Thus, the effect of inappropriate selection of parameters can be minimized.

2.2 Adaptive Morphological Filter

The basic concept of our algorithm is based on the theory of mathematical morphology. In our algorithm, the whole point dataset is assumed to be initially scanned by a small line-like window $m * n$. The intention of using small windows is to preserve as much detail as possible in slope areas. And then the window size is changed adaptively during the course of data processing. The window will move forward gradually to scan over the whole area, and the process will repeat along four directions i.e. left to right, right to left, top to bottom and bottom to top. Then an intersection of results from all processes is obtained. The reason for the four-direction processes is to avoid errors created by complex building structures. Figure 1 illustrates an example of complex building structures which will result in misclassification with single direction scanning. Since the proposed adaptive filter attempts to find elevation rise and drop to detect the length of a building, it

leads to such kind of errors when the scanning window hits multi-rooftop buildings. Sometimes high vegetation and low buildings can also create similar circumstances. Such errors can be significantly reduced by using the four-direction processes.

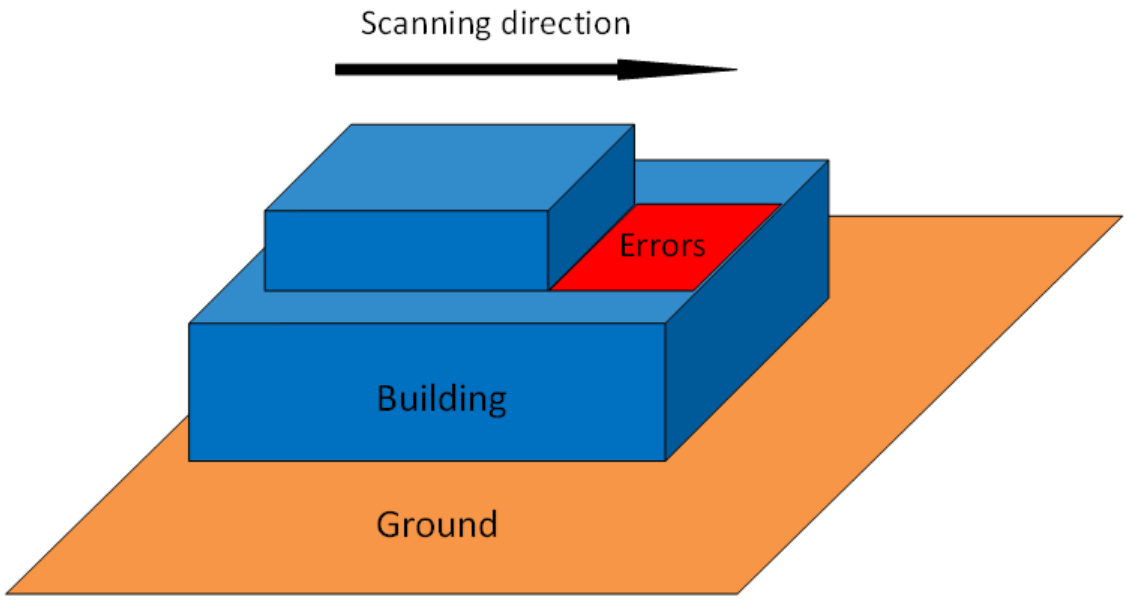


Figure 1. An illustration of errors resulting from complex building structure

During the process, for each window w_d, the points with the lowest and highest elevation values are firstly detected. Every point in w_d that falls within a threshold h_t above the lowest point is classified as a ground point. Then the window moves a small distance along the scanning direction and repeats the process until the whole dataset is scanned. The adaptive window-size indicator flags when there is a height difference larger than a threshold within the window. The hypothesis of this adaptive method is that a set of points on and around a building can contain a significant elevation rise or drop. Thus an approximate size of the building can be detected by measuring where the elevation increases or decreases, and the window size can be changed accordingly. The adaptive window-size indicator is implemented by the following operations: The height difference H_d within w_d is first calculated. If H_d is larger than a predefined threshold h_b, the algorithm will consider that there are parts of buildings within the window and then check the elevation of the points that fall ahead of w_d. This checking process uses a smaller checking window w_f ($I_m * n$) which is ahead of w_d to detect elevation change (Figure 2). The process will stop until there is a large elevation decrease or the supposed building is larger than a predefined maximum building length. The scanning window size will consequently increase to cover the detected building or stay still according to the previous results. The elevation change during the detection of a building size is expressed as,

$$\Delta_f = e_{w_{f-1}} - e_{w_f} \quad \text{for } f=1,2,3\ldots \tag{1}$$

When $\Delta_f \geq h_b$ it is considered that the window reached the other edge of the building and the window size will increase to $m + f * I_m$ in order to cover the whole building. On the other hand, if $I_m > \max(L_b)$, the process will be forced to stop and the default window size will be used, where $\max(L_b)$ is the predefined largest length of a building in the study area. After the increased window

scans over the building, the window size will be set to the default value and repeat the adjusted window-based detection (Figure 3). When the ground points are filtered, all the points of 0.3 m higher than ground are classified as above-ground features, where 0.3 m is the vertical accuracy of the data used in this study.

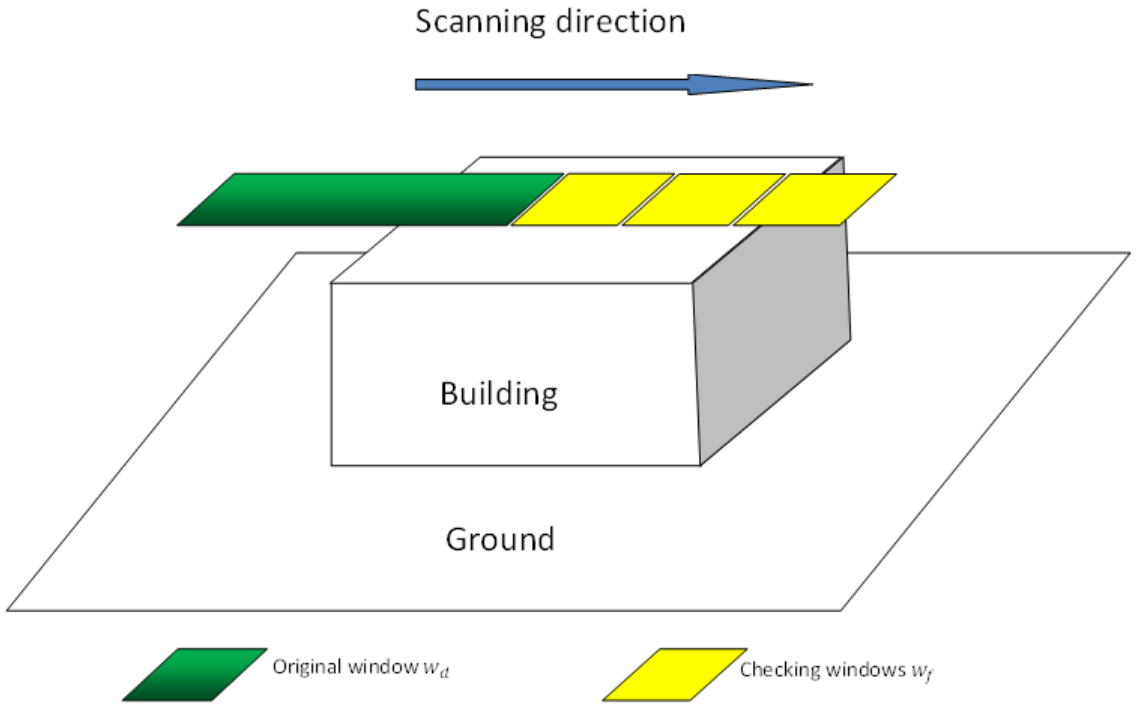


Figure 2. An illustration of scanning windows

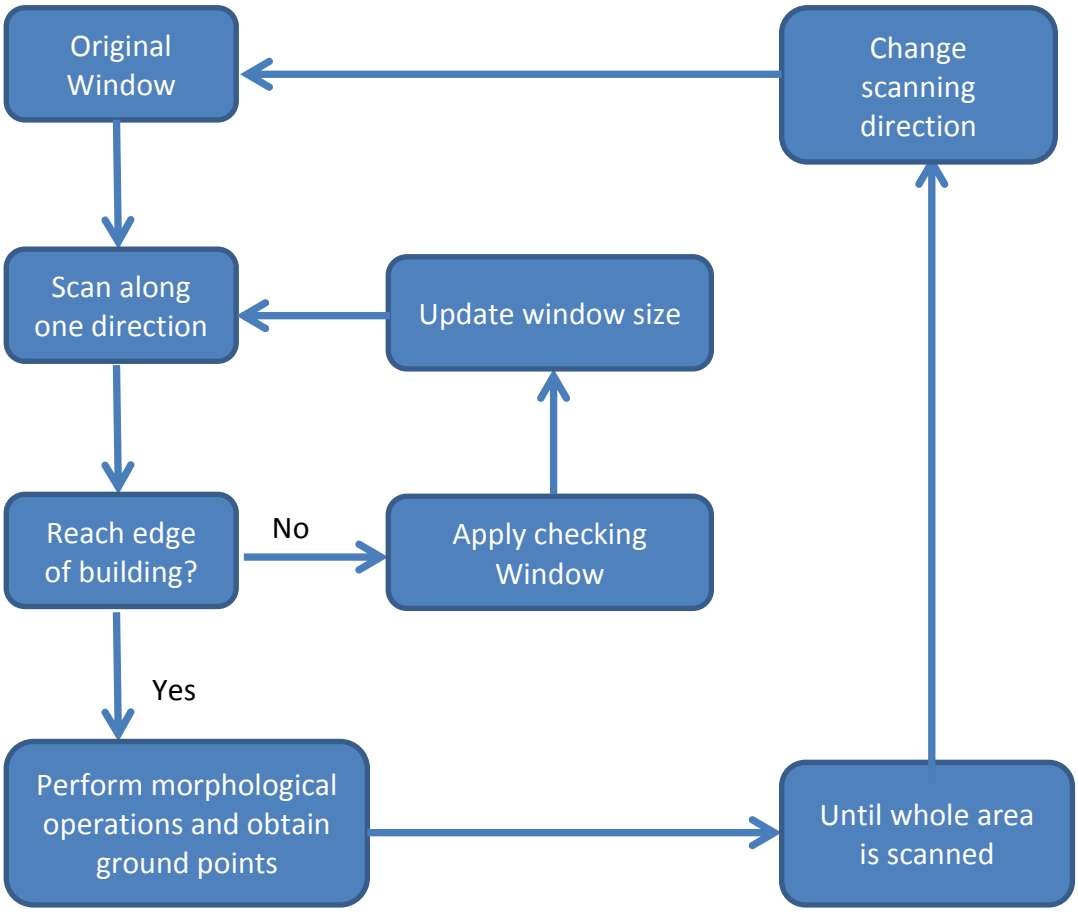


Figure 3. Flow chart of the scanning process

3. Vegetation Removal

In urban areas, vegetation is usually the main feature besides buildings. Other features may include bridges, cars, towers and small artificial structures. It is important to remove those unwanted objects after ground filtering. Normalized Difference Vegetation Index (NDVI) is commonly used to detect healthy vegetation and thus to remove it (Aytekin, Ulusoy et al. 2009, Awrangjeb, Ravanbakhsh et al. 2010). However, the availability of multispectral images and seasonal changes of trees limit the usage of this method. In addition, NDVI may not achieve a reliable accuracy if the choice of thresholds is inappropriate. Thus, residuals often remain after the process of the NDVI method.

We used a progressive approach after applying NDVI to remove the residuals as well as other unwanted small features. The hypothesis is that these unwanted measurements usually tend to be scattered and cannot form a large and regular cluster of points representing a building. Then the hierarchical clustering based on Euclidean-Distance is performed to the points. The three-dimensional distances between points rather than two-dimensional are calculated for clustering. Then a progressive approach including height-, areas- and number-of-points-based thresholding was applied step by step to remove the clusters that do not contain building points. Firstly, clusters of

average above-ground elevation lower than 2 m, and also small clusters that have less than 20 points i.e. isolated points, lower sparse vegetation points, were removed. Then the alpha-shape algorithm was applied on each cluster, and corresponding boundary polygons of the clusters are obtained. The main purpose of this step is to remove those features that are small but contain a decent amount of points, such as cars, small shed or stairs. Therefore, a minimum area threshold is used. The threshold may vary depending on the local condition. For example, area-threshold should be larger in a metropolitan area which consists of relatively larger buildings than that in a residential area which mainly consists of relatively smaller buildings. Consequently, a threshold of 30 square meters is assigned for our study areas.

4. Building Boundary Extraction

There are several algorithms for extraction of polygons from points (Sampath and Shan 2007, Wei, Jin et al. 2011, Lee and Kim 2012). In this study, the alpha-shape algorithm is applied on the classified building point clusters to detect the points that lie on the boundary. The topology information of these points will be preserved; i.e. adjacent points in the point cloud are also stored in a certain order in the output matrix (Figure 4). The α value used in this study is 1.5 m, which is between the average point spacing and twice the point spacing of the point data.

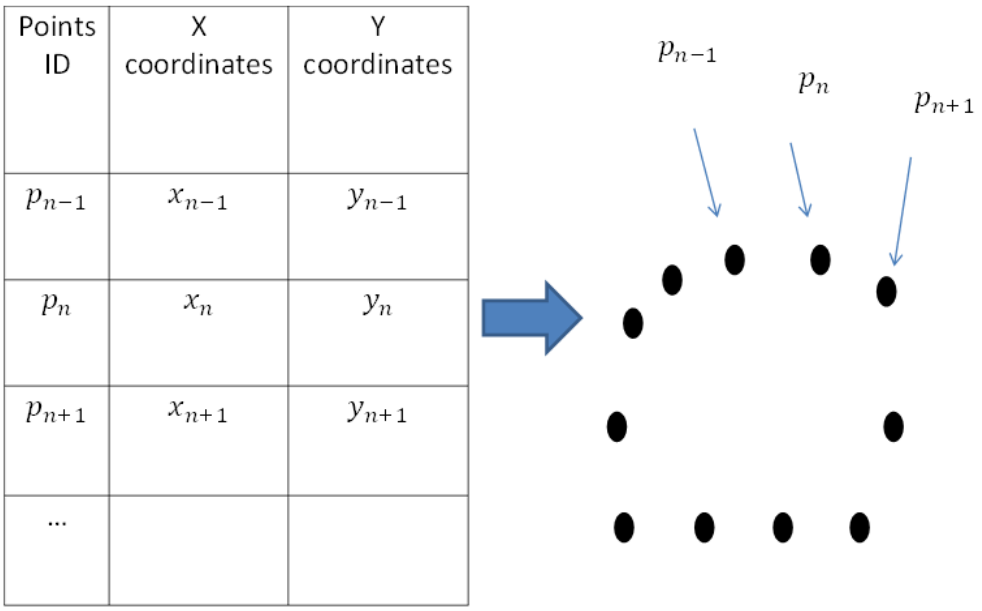


Figure 4. Relationship between the output matrix and the point cloud

Various algorithms exist to delineate zigzag boundaries or curves. The Douglas–Peucker algorithm is one of the most well-known algorithms. In the classical Douglas–Peucker algorithm, essential vertices are preserved and intermediate points are removed. As a result, simplified lines can be formed with the preserved vertices. However, the original algorithm is designed for simplification of lines or paths and thus may not be suitable for simplification of polygons. In the case of simplification of lines, the algorithm tends to keep the starting and ending points as essential vertices. However, the starting and ending points on the boundary of a polygon are very close to each other. That is, it is not reasonable to keep both of them. We keep only the starting points and delete the ending points. A possible consequence of this method is that one edge may be divided

into two when the starting point lies in the middle of an edge. However, such a case has a minor effect on the shapes or areas of the delineated polygons and thus can be ignored. The original direction of the boundary lines is preserved since it can show the quality of our building detection approach and can be assessed in the evaluation section. In addition, it is possible to create more errors during the process of changing the direction of boundary lines. The tolerance we used in the Douglas–Peucker algorithm is same as the α value in the alpha-shape algorithm since it is the largest neighbourhood searching radius in those algorithms. Although a smaller tolerance may reduce the number of self-intersecting polylines, it also increases the roughness of the boundary lines. The results of the extracted polygons are shown in the evaluation section, and the performance is assessed and discussed.

5. Performance Evaluation

5.1 Datasets

The lidar data provided by Land and Property Information (LPI), New South Wales (NSW), Australia, was acquired over Bathurst, NSW, on 20 April 2011 with a commercial laser scanner Leica ALS50-II. The data is provided in LAS format containing up to 4 returns for each pulse. Multiple returns usually occur on the edge of buildings or trees that allow the laser beam to penetrate. The horizontal and vertical accuracy of the lidar data is 0.8 m and 0.3 m, respectively, with 95% confidence. The data has an average point density of 1.57 points per square meter. The processed data has been manually edited by LPI to meet the classification level 3 whereby the ground class contains minimal non-ground points such as vegetation, water, bridges, temporary features, jetties etc. The aerial image is obtained on 10 April 2013. The ortho-image contains 4 bands including red, green, blue and infrared spectrum. Reference building polygons are digitized from this image and are used to assess the test results.

5.2 Evaluation method

The evaluation process is divided into three categories including object-based evaluation, area-based evaluation, and t-test comparison. The assessment is performed on two sites: Site A and Site B. For the object-based evaluation, the number of detected buildings and reference buildings are counted and used for the analysis. For the area-based evaluation, areas of the building polygons are used for all detected and reference entities. A paired t-test is also carried out in order to compare the difference between the digitized reference data and our results.

5.2.1 Object-based evaluation

Five indices are used for the object-based evaluation to evaluate the number of buildings counted in both extracted data and reference data. Completeness C_m, correctness C_r, quality Q_i, detection fusion rate D_u and detection fission rate D_i are adapted from Awragjeb et al. (Awrangjeb, Ravanbakhsh et al. 2010). Originally, these indices were calculated by the matched pixels and the total number of pixels. In this semi-automatic processing, the equations are modified to represent the object-based accuracy of the extracted polygons as follows:

$$C_m = ref_c / ref_n \quad (2)$$

$$C_r = ext_c / ext_n \quad (3)$$

$$Q_i = (C_m * C_r)/(C_m - C_m * C_r + C_r), \quad (4)$$

where $\mathrm{ref_c}$ is the number of matched polygons in the reference data, $\mathrm{ext_c}$ is the number of matched polygons in the extracted polygons, $\mathrm{ref_n}$ is the total number of buildings in the reference and $\mathrm{ext_n}$ is the total number of buildings in the extraction.

The remaining two are defined as:

1. Detection fusion rate is the percentage of overlapped polygons where a single polygon in the extracted result represents multiple polygons in the reference. It is defined as

$$D_u = (ref_m - ext_s)/ref_n, \quad (5)$$

where $\mathrm{ref_m}$ is the number of overlapped multiple polygons in the reference and $\mathrm{ext_s}$ is the number of single polygons in the extraction.

2. Detection fission rate is defined as the percentage of overlapped polygons where multiple polygons in the extracted result represent a single polygon in the reference. It is expressed as

$$D_i = (ext_m - ref_s)/ext_n, \quad (6)$$

where $\mathrm{ext_m}$ is the number of overlapped multiple polygons in extraction and $\mathrm{ref_s}$ is the number of corresponding single polygons in the reference.

Table 1 shows the object-based statistics, while Tables 2-3 show the number of buildings of different classes. Ref 1 and Ext 1 refers to the reference data and the extracted results in Site A respectively, while Ref 2 and Ext 2 refers to those in Site B. The total number of buildings of area less than 30 $\mathrm{m^2}$ in Site A is three which are exactly those unmatched buildings, while the number in Site B is also three which are one less than the unmatched buildings.

Table 1. Object-based assessment

Object-based	C_m	C_r	Q_i	D_u	D_i
Site A	96.34%	98.46	94.91%	21.95%	4.62%
Site B	94.29%	92.73%	87.80%	21.43%	0.00%

Table 2. Object-based statistics in Site A

	Total no.	Fusion	Fission	No. of overlaps	No. of non-overlaps	unmatched	matched
Ref 1	82	30	3	33	49	3	46
Ext 1	65	12	6	18	47	1	46

Table 3. Object-based statistics in Site B

	Total no.	Fusion	Fission	No. of overlaps	No. of non-overlaps	Unmatched	matched
Ref 2	70	23	0	23	47	4	43
Ext 2	55	8	0	8	47	4	43

5.2.2 Area-based evaluation

For the area-based evaluation, areas of polygons are used for all detected and reference buildings. A total of five area-based indices are used, containing completeness ($\mathrm{C_{ma}}$), correctness ($\mathrm{C_{ra}}$), quality ($\mathrm{Q_{ia}}$), area omission error ($\mathrm{E_{ro}}$) and area commission error ($\mathrm{E_{rc}}$).

These indices are calculated similarly as in the object-based evaluation. The intersection of extracted polygons and reference polygons is considered as correctly detected areas. Thus, these indices can be obtained as follows:

$$E_{rc} = (ext_a - int_a)/ext_a \quad (6)$$
$$E_{ro} = (ref_a - int_a)/ref_a \quad (7)$$
$$C_{ma} = int_a/ref_a \quad (8)$$
$$C_{ra} = int_a/ext_a \quad (9)$$

where $\mathrm{ext_a}$ is the total area of buildings in the extracted polygons, $\mathrm{int_a}$ is the total area of buildings in the matched/intersecting polygons, and $\mathrm{ref_a}$ is the total area of buildings in the reference polygons. Table 4 shows the statistics of this evaluation method.

Table 4. Area-based assessment

Area-based	C_{ma}	C_{ra}	Q_{ia}	E_{rc}	E_{ro}
Site A	88.28%	91.35%	81.47%	8.65%	11.72%
Site B	86.32%	88.83%	77.87%	11.17%	13.68%

5.2.3 T-test comparison

The t-test is suitable for determining the level of significance of difference of two sets of data; i.e., our extracted polygons and the reference polygons. If the null hypothesis cannot be rejected at a certain level, there is no significant difference between the extracted results and the reference on that level. In our test the area is analysed using a two-tail paired t-test, as the areas of the extracted polygons can either be greater or less than those of the reference. We only use matched polygons in the paired test. If one polygon in one dataset overlaps multiple polygons in the other dataset, we merge the attributes of the multiple polygons in order to perform the paired test. By doing so, there are 61 and 51 pairs to be compared finally in Site A and Site B, respectively.

The formula to calculate the t value is,

$$t = (\bar{x} - \Delta)/(s/\sqrt{n}), \quad (7)$$

where $\bar{x}$ is the mean of the difference between two datasets, Δ is the hypothesized difference, s is the standard deviation of the differences, and n is the data size. The number of degree of freedom for this test is n-1. We set Δ to 0, and the null hypothesis is that there is no difference between the two paired areas in terms of the mean difference. We use a significant level of 0.01 and 0.05; i.e. $\alpha = 0.01$ or $\alpha = 0.05$. Using Equation (7), we can obtain the t value.

$$t_{Site\ 1} = 7.18/(36.65/\sqrt{61}) = 1.5301$$
$$t_{Site\ 2} = 10.71/(40.32/\sqrt{51}) = 1.8970$$

By checking the t distribution critical-values table, we find

$$t_{df=60,\alpha=0.01} = 2.6649$$
$$t_{df=50,\alpha=0.01} = 2.6778$$
$$t_{df=60,\alpha=0.05} = 2.0025$$
$$t_{df=50,\alpha=0.05} = 2.0086$$

Because the computed t values of 1.5301 and 1.8970 do not exceed 2.0025 and 2.0086, respectively, the null hypothesis cannot be rejected at the 0.05 level. We can also calculate the p value using the computed t value and the number of degrees of freedom.

$$p_{Site\ 1} = 0.1312$$
$$p_{Site\ 2} = 0.0636$$

Since $\alpha \leq p$, statistically, we can accept both of the null hypotheses at the 0.05 level. The null hypothesis of Site A can also be accepted at the 0.10 level. Overall, our method does not produce a significant error on the area of building polygons compared to the digitized reference.

5.3 Experiment Results and Discussion

This experiment was implemented under Window 7 Enterprise SP1 using Matlab R2011b and ArcGIS 10.0. The process was semi-automatic i.e. the result for each step is obtained automatically while data type conversion and data transmission between different platforms is done manually. The aerial images and digitized references of building polygons can be seen in Figures 5-6. Samples of the extracted polygons in our test are shown in Figures 7-8.

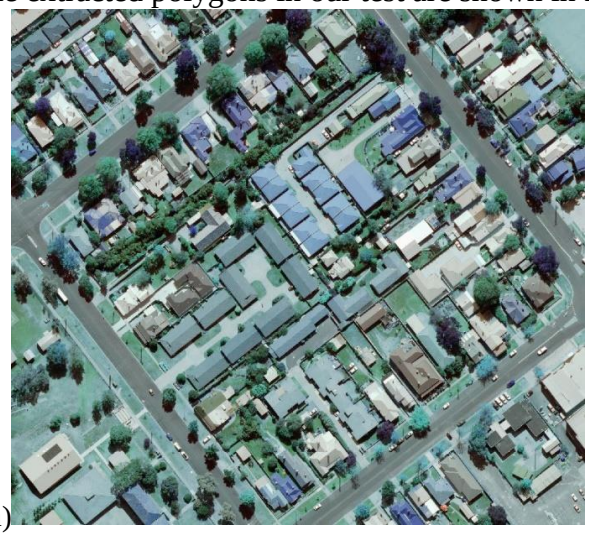
(a)  (b) 

Figure 5. Site A: (a) the aerial image, and (b) the digitized polygons

(a)  (b) 

Figure 6. Site B: (a) the aerial image, and (b) the digitized polygons

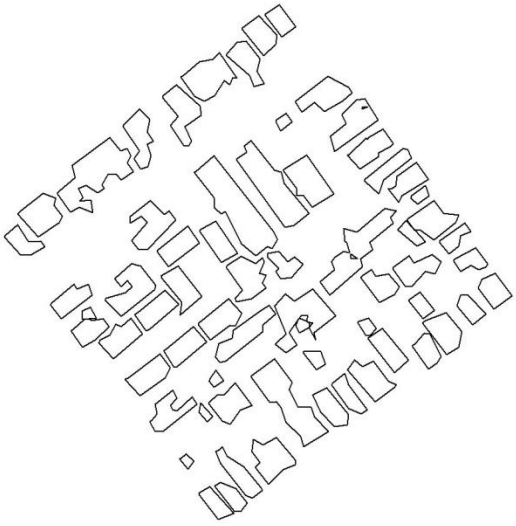
(a)  (b) 

Figure 7. Site A: (a) the extracted polygons superimposed on the aerial image, and (b) the extracted polygons

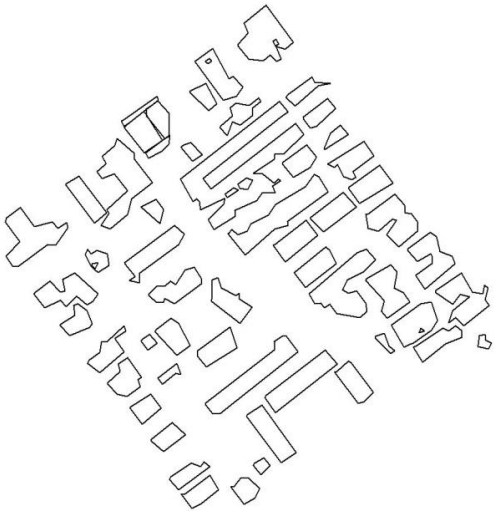
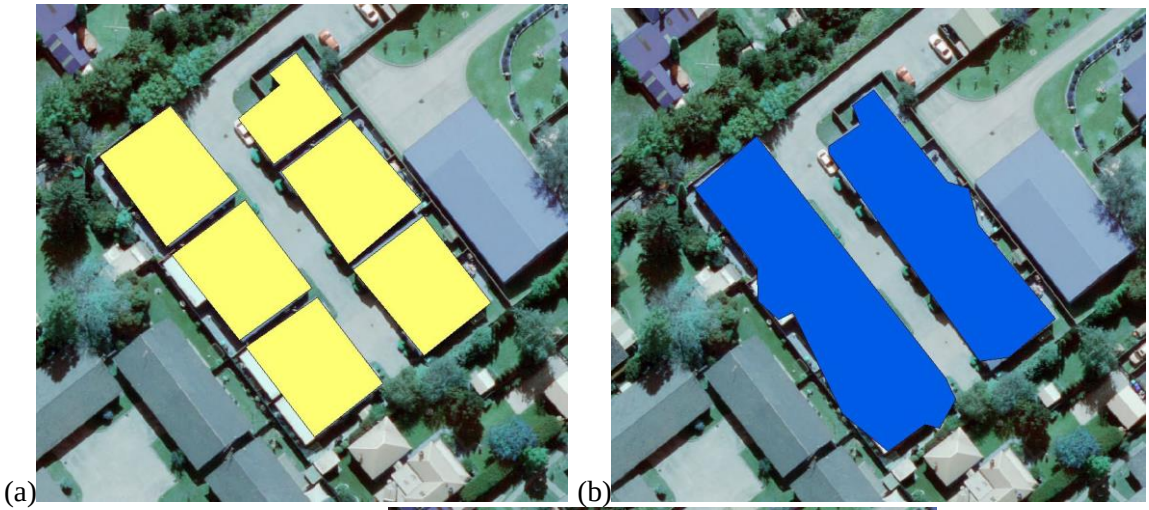
(a)  (b) 

Figure 8. Site B: (a) the extracted polygons superimposed on the aerial image, and (b) the extracted polygons

(a) (b)



(c)

Figure 9. Single-multiple overlap areas: (a) the reference polygons in the single-multiple overlap areas, (b) the extracted polygons in the single-multiple overlap areas, (c) the single-multiple overlap areas

As can be seen from Figure 9, a single polygon in the extracted result actually contains multiple polygons in the reference. We will call this phenomenon 'single–multiple overlaps'. The aerial image (Figure 9c) shows that the gap between each building is very small (less than 1.5 m) and all the buildings are of similar heights. In our approach, different buildings were separated using hierarchical clustering based on a distance threshold; i.e., points that are close to each other are clustered as parts of the same building. Therefore, it is very difficult to separate these buildings within the lidar point cloud using a semi-automatic process. As a result, it will be difficult to perform the paired t-test and assess the object-based accuracy. What is done to resolve this problem is to merge the attributes (areas) of the multiple polygons with the geometrical characteristics preserved. In that case, when the paired t-test is conducted, the area of the single polygon in the extracted results is matched with the total area of three corresponding polygons in the reference. The errors caused by missing gaps, therefore, can be taken into account statistically. For the object-based evaluation, the number of undetected buildings rather than the detected ones is counted in order to obtain the completeness (C_m). However, this index still contains the error caused by single–

multiple overlaps. Therefore, we introduce the detection fusion rate and detection fission rate to represent such an error. A more statistically strict completeness can be derived using $C_m - D_u$, while a more strict correctness can be calculated using $C_r - D_i$. Because these single–multiple overlaps are indeed detected due to the errors caused by the limitation of the lidar point cloud, it is suggested that the multiple polygons can be successfully detected. In the area-based evaluation, the single–multiple overlaps do not have an impact on the analysis.

(a)  (b) 

Figure 10. Misidentification: (a) an example of misidentification, and (b) the original aerial image

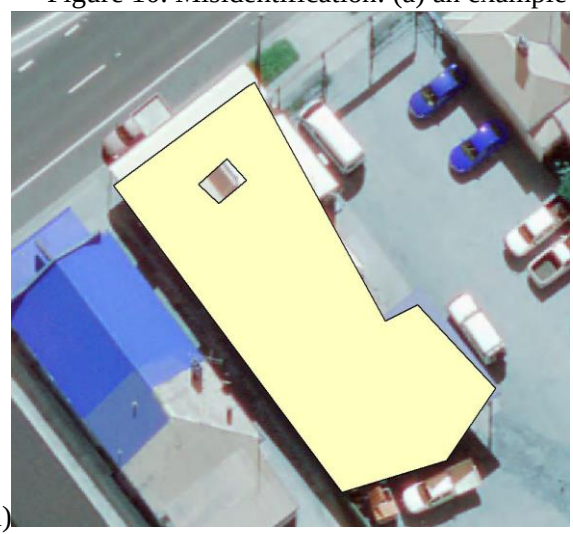
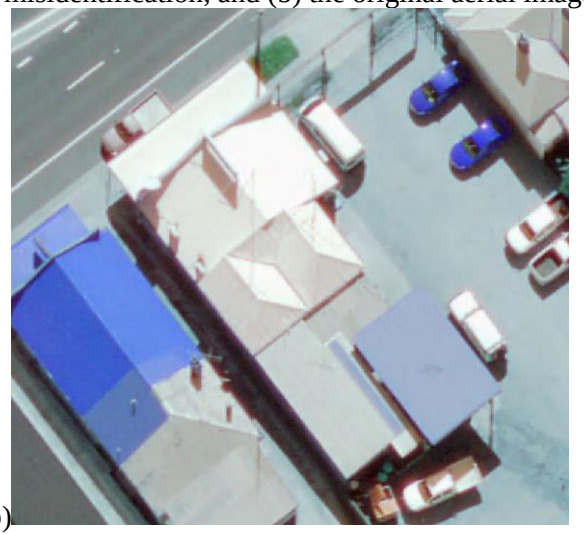
(a)  (b) 

Figure 11. Omission error: (a) an example of omission error due to the spike on the rooftop, and (b) the original aerial image

In the t-test, the similarity of the paired polygons excluding the undetected ones is examined. The area-based assessment shows the overall quality of the results, while the t-test shows how close the extracted polygons are to the reference ones. The null hypothesis of both sites can be supported at the 0.05 level, while that of Site A can also be accepted at the 0.1 level. That is, the results are statistically very similar to the reference.

However, there are some omission and commission errors; i.e., unmatched polygons. An example of commission errors can be seen in Figure 10. In Figure 10, a building-like shed is misclassified as a building. Such misclassification is a critical problem since it is very difficult to distinguish those sheds from buildings with x,y,z coordinate information only, which is the limitation of lidar data. Another typical omission error can be seen in Figure 11. As we apply hierarchical clustering based on distance to separate buildings points and isolated points, if there is a spike that is much higher than the rooftop, the spike will be considered as residuals and removed. As a result, there will be small holes inside the boundary polygons.

Overall, with all the assessments conducted, our semi-automatic approach is able to provide a reliable building extraction results. Future work may include separation of the single–multiple overlaps with the assist of an image-processing method.

6. Concluding Remarks

A semi-automatic approach to building extraction is proposed to take advantage of the point-based processing methods. Three steps are taken in the approach: ground filtering, vegetation removal, and building boundary delineation. In general, a progressive strategy is applied to detect and extract the buildings. That is, the unwanted points are classified and removed gradually rather than building points are obtained directly. First, an adaptive morphological filter is used to separate ground and non-ground features. The filter is designed to work directly on the lidar points. During the filtering process, the whole point cloud is scanned with a moving window in four directions. The size of the window is chosen automatically depending on the local situation. In the second step, a fusion of NDVI, thresholding and hierarchical clustering is applied to remove vegetation and the residuals. The majority of vegetation is initially detected by assigning NDVI values to all the non-ground points. Then the residuals are extracted by combining the clustering and thresholding methods based on the hypothesis that the residuals tend to be isolated or scattered. As a result, small clusters are removed and the majority of building points are preserved. In the final step, the boundary polygons are delineated using alpha shape and Douglas-Peucker algorithms. During the process, all the topology information is preserved in order to create polygons from points. The quality of the simplified polygons is assessed using the reference polygons digitised from the aerial images. Overall, the extracted building polygons are shown to be reliable with approximately 90% object-based accuracy and 80% area-based accuracy. The t-test indicates that the correctly-extracted polygons are of great similarity to the reference data.

It is demonstrated that this approach is able to create accurate boundary polygons with acceptable errors. Typical errors usually occur when there are insufficient boundary points due to overlaps and over-removal or when there are misclassified points which are difficult to distinguish from buildings. It is difficult to detect the actual shape of a building when the building is blocked or covered by a huge tree crown and there are almost no lidar points falling on the rooftop, which is the shortcoming of lidar data. It might possibly be solved by a fusion of different data types.

There are several possible opportunities for future research that can be built on this study. The efficiency of the filtering algorithm can be enhanced by using other programming languages, or by

improving current coding. Moreover, high-resolution images can play a more important role in the building detection process, as the proposed method tends to preserve objects of similar attributes with buildings when lidar data is used as the main input data source. However, this improvement will require a fusion of image processing and point processing, rather than a point-based method alone. Another possible improvement is the extension of this study to a 3D scale through the inclusion of multiple data sources. Studies on the 3D reconstruction of rooftops can create more detailed boundaries and produce better visual representations of the buildings.

REFERENCES

Akkiraju, N., H. Edelsbrunner, M. Facello, P. Fu, E. P. Mcke and C. Varela (1995). Alpha Shapes: Definition and Software. Proceedings of the 1st International Computational Geometry Software Workshop.

Awrangjeb, M., M. Ravanbakhsh and C. S. Fraser (2010). "Automatic detection of residential buildings using LIDAR data and multispectral imagery." ISPRS Journal of Photogrammetry and Remote Sensing 65(5): 457-467.

Awrangjeb, M., C. Zhang and C. S. Fraser (2013). "Automatic extraction of building roofs using LIDAR data and multispectral imagery." ISPRS Journal of Photogrammetry and Remote Sensing 83(0): 1-18.

Axelsson, P. (1999). "Processing of laser scanner data—algorithms and applications." ISPRS Journal of Photogrammetry and Remote Sensing 54(2–3): 138-147.

Axelsson, P. (2000). "DEM generation from laser scanner data using adaptive TIN models." International Archives of the Photogrammetry and Remote Sensing XXXIII: 110-117.

Aytekin, O., I. Ulusoy, A. Erener and H. S. B. Duzgun (2009). Automatic and unsupervised building extraction in complex urban environments from multi spectral satellite imagery. 2009 4th International Conference on Recent Advances in Space Technologies (RAST 2009), 11-13 June 2009, Piscataway, NJ, USA, IEEE.

Dash, J., E. Steinle, R. P. Singh and H. P. Bähr (2004). "Automatic building extraction from laser scanning data: an input tool for disaster management." Advances in Space Research 33(3): 317-322.

Dong, H. L., M. L. Kyoung and U. L. Sang (2008). "Fusion of lidar and imagery for reliable building extraction." Photogrammetric Engineering and Remote Sensing 74(2): 215-225.

Forlani, G., C. Nardinocchi, M. Scaioni and P. Zingaretti (2006). "Complete classification of raw LIDAR data and 3D reconstruction of buildings." Pattern Analysis and Applications 8(4): 357-374.

Huang, X. and L. Zhang (2011). "A multidirectional and multiscale morphological index for automatic building extraction from multispectral GeoEye-1 imagery." Photogrammetric Engineering & Remote Sensing 77(7): 721-732.

J. Kilian, N. H., M. Englich (1996). "Capture and Evaluation of Airborne Laser Scanner Data." International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXI-B3: 383-388.

Keqi, Z., Y. Jianhua and C. Shu-Ching (2006). "Automatic Construction of Building Footprints From Airborne LIDAR Data." Geoscience and Remote Sensing, IEEE Transactions on 44(9): 2523-2533.

Lee, J.-H. and Y.-I. Kim (2012). "Building boundary reconstruction from airborne lidar data by adaptive convex hull algorithm." Journal of the Korean Society of Surveying Geodesy Photogrammetry and Cartography 30(3): 305-312.

Mahmoud, S. and J. Trinder (2010). Support Vector Machines Based Filtering of LiDAR Data: A Grid Based Method. FIG Congress 2010. Sydney, Australia.

Meng, X., N. Currit and K. Zhao (2010). "Ground Filtering Algorithms for Airborne LiDAR Data: A Review of Critical Issues." Remote Sensing 2010(2): 833-860.

Miliaresis, G. and N. Kokkas (2007). "Segmentation and object-based classification for the extraction of the building class from LIDAR DEMs." Computers and Geosciences 33(8): 1076-1087.

Potsiou, C. (2010). Rapid Urbanization and Mega Cities: The Need for Spatial Information Mangement. Copenhagen, Denmark.
Sampath, A. and J. Shan (2007). "Building boundary tracing and regularization from airborne lidar point clouds." Photogrammetric Engineering and Remote Sensing 73(7): 805-812.
Sohn, G. and I. Dowman (2007). "Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction." ISPRS Journal of Photogrammetry and Remote Sensing 62(1): 43-63.
Tan, Q. and J. Wang (2011). "Building extraction using high resolution multi-spectral image and LiDAR data." Yingyong Jichu yu Gongcheng Kexue Xuebao/Journal of Basic Science and Engineering 19(5): 741-748.
Wei, S., Z. Jin and Y. Feng (2011). A new algorithm of building boundary extraction based on LIDAR data. 2011 19th International Conference on Geoinformatics, 24-26 June 2011, Piscataway, NJ, USA, IEEE.
Zhang, K., S.-C. Chen, D. Whitman, M.-L. Shyu, J. Yan and C. Zhang (2003). "A progressive morphological filter for removing nonground measurements from airborne LIDAR data." Geoscience and Remote Sensing, IEEE Transactions on 41(4): 872-882.