A proposed new approach for extracting complex building boundaries from segmented LiDAR data

Jeong-Ho Lee, Soo-Hee Han, Young-Gi Byun, Kiyun Yu, Yong-Il Kin
School of Civil, Urban and Geosystem Engineering
Seoul National University
38-223, Seoul National University, Shinrim-Dong, Kwanka-Ku, Seoul 151-742
REP. of Korea
ilome79@snu.ac.kr, scivile2@snu.ac.kr, kko071@snu.ac.kr, yik@snu.ac.kr, kiyun@snu.ac.kr
http://spins.snu.ac.kr

Abstract: - Many studies have been conducted on extracting building boundaries from LiDAR (Light Detectino And Ranging) data. In these studies, points are first segmented, then are further generalized or regularized to acquire straight boundary lines that better approximate the real boundaries. In most research in this area, such generalization or regularization assumes that buildings have only right angles, i.e., all the line segments of the building boundaries are either parallel or perpendicular to each other. However, this assumption is not valid for many buildings, and restricts wider applications of this type of study. We present a new approach consisting of three steps that is applicable to more complex building boundaries. The three steps consist of boundary tracing, generalization, and regularization. Each step contains algorithms that range from slight modifications of conventional algorithms to entirely new concepts. Four typical building shapes were selected to test the performance of our new approach. Our results show that our proposed approach has good potential for extracting building boundaries of various shapes.

Key-Words: - LiDAR, boundary tracing, generalization, regularization, building boundary

1 Introduction

Recently, LiDAR has become more and more widely used for generating Digital Elevation Models (DEM) and detecting building boundaries [1] [5] [7] [8] [10] [11] [12]. In many cases, LiDAR has replaced traditional satellite and aerial photo-based feature extraction [4] [9] [14]. This is because LiDAR renders positional data with a very high accuracy as well as it is active in sensing, allowing for weather-free and time-flexible data collection.

In one of the most popular applications of LiDAR, building boundary extraction, processes such as filtering are often carried out to generate DEM from raw data, followed by segmentation and sometimes classification [1] [6] [10] [11]. Once segmentation is complete, some additional steps are usually required to acquire more accurate and realistic boundary shapes [7] [12]. In fact, segmented building boundaries are seldom clean, but are zigzag shaped and their corners are by and large inaccurate, thus, requiring further processes such as generalization or regularization. So far, a series of research activities related to this issue has been carried out.

Weidner and Forstner used the Minimum Description Length (MDL) method to regularize ragged building boundaries [15]. Alharthy and Bethel presented a polygon extraction algorithm that estimated the dominant directions using an angle histogram [1]. This algorithm was also applied under the constraint that buildings have only two dominant directions. Ma classified line segments into two groups which were supposed to be perpendicular. Thus, a weighted adjustment was carried out to calculate the azimuth of the two classes, after which the segments were adjusted and grouped [11]. Sampath used the least squares method under the conditional constraint that the slopes of the parallel lines are equal and the product of the slopes of the perpendicular lines is -1 [12].

Previous researchers have tried to either generalize or regularize building boundaries by assuming that most buildings have only right angles, i.e., that all the line segments of building boundaries are either parallel or perpendicular to each other. Under this assumption, researchers have applied their proposed algorithms only to rectangular buildings. However, in reality, buildings usually exhibit complex shapes, including non-right angle edges. To work on such buildings, new algorithms need to be developed.

In this paper, we present a new approach consisting of three steps that is applicable to buildings that have complex boundary shapes. The proposed three steps are: trace of boundaries, generalization, and regularization. These stages are explained in detail, and the paper is organized as follows. In Section 2, an efficient method for tracing boundaries that is a modification of the convex hull algorithm is explained. The Section 3 describes the technical details for the boundary generalization in which the conventional Douglas-Peucker algorithm is partially used. The Section 4 shows a new boundary regularization process in which the least squares method is adopted. To apply our proposed approach, four buildings were selected whose boundary shapes are often detected in many urban areas, and the resulting accuracy is examined. This is explained in Section 5. Finally the discussions and future research directions are contained in Section 6.

2 Trace of Boundaries

To trace boundaries, the raw data points have to be segmented in advance. Because many researchers have been working on segmenting raw data points [12] [13], we do not delve deeply into this topic. Thus, assuming that the data points have already been segmented, the trace begins with an analysis of the convex hull algorithm.

The convex hull algorithm determines the smallest convex set containing discrete points. However, many buildings with a complicated shape are not convex. Applying the convex hull algorithm directly to these buildings cannot lead to an accurate determination of the shape of the building, nor can it contain all the boundary points.

Therefore, a modified version of the convex hull algorithm is proposed to overcome such problems. We restrict the search space of the convex hull algorithm to a shorter distance by constructing grids for the points in a building. For each grid cell to contain enough number of points, the grid size is set to double the mean distance between two points. Thus, the search space is restricted to the points in the current cell (i.e., that containing the previous detected boundary point) and the next cell. The detailed process to trace boundary points is given below.

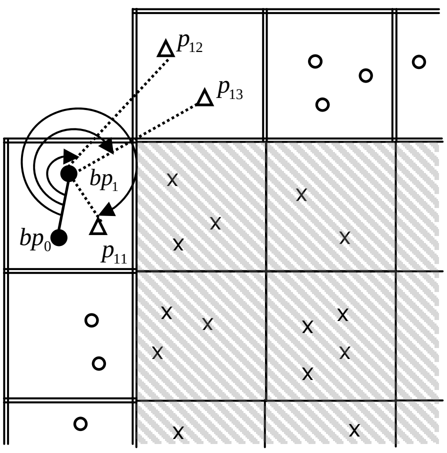


Fig.1 Boundary tracing- Finding the minimum clockwise angle

Step 1. For each building, create a grid and detect the starting boundary point bp_0 which has a minimum x-coordinate of all the points in a building.

Step 2. Select the outer cells in clockwise order by starting from the cell containing bp_0 . This is conducted by using a chain code.

Step 3. Select the points $P_o = \{p_{01}, p_{02}, \cdots, p_{0m}\}$ within a given distance from bp_0, i.e., the points in the current cell containing bp_0 and the next cell.

Step 4. Determine the slopes of all the lines that connect the point bp_0 with the points selected in Step 3.

Step 5. Find the next boundary point bp_1 in set P that has the least clockwise angle from the y axis.

Step 6. Select the set of points P_1 within a given distance from bp_1 and determine all the lines that connect the point bp_1 with the points in set P_1. In the case shown on Fig. 1, $P_1 = \{p_{11}, p_{12}, p_{13}\}$

Step 7. Find the next boundary point bp_2 in set P_1 that has the least clockwise angle from the line $\overline{bp_1\ bp_0}$ to the line $\overline{bp_1\ p_i}$. In case shown on Fig.1 , p_{12} is found for bp_2 .

Step 8. Continue until the starting boundary point bp_0 is found.

By constructing a grid and restricting the search space to the points in the current cell and the next cell, it is possible to trace building boundary points not only for convex shape but also for non-convex shape. In addition, constructing a grid and having the correspondence relationship between a cell and the points in it helps finding points quickly and easily from many irregularly distributed points. This is because cells are stored in computer's memory in good order (clockwise). In addition, restricting the search space increases the speed of the convex hull algorithm.

3 Generalization

After tracing the boundary, generalization is conducted to form groups of points belonging to each line segment. Because the edge lengths of buildings can vary, there are limitations in applying the conventional generalization algorithms [2] [3] directly to building boundaries. Therefore, we generalize boundaries using a two-stage procedure based on distance and angle. In the first stage, a conventional Douglas-Peucker algorithm based on orthogonal distance is used. In the second stage, by considering the peculiarity of buildings, we use the constraints of orthogonal distance, angle, and minimum length of edge. It is possible to further remove redundant points from the first corner points selected in the previous stage. The process of two-step generalization is as follows.

Step 1. Determine the four initial points that have extreme coordinate values: a point with a minimum y-coordinate, a point with a maximum y-coordinate, a point with a minimum x-coordinate, and a point with a maximum x-coordinate

$\{cp_{x\_\min}, cp_{x\_\max}, cp_{y\_\min}, cp_{y\_\max}\}$

Step 2. Apply the Douglas-Peucker algorithm recursively to all the boundary points between two consecutive initial points, and find the first corner points that have a longer orthogonal distance than H_1.

$first\ set = \{cp_1, cp_2, \cdots, cp_k\}$

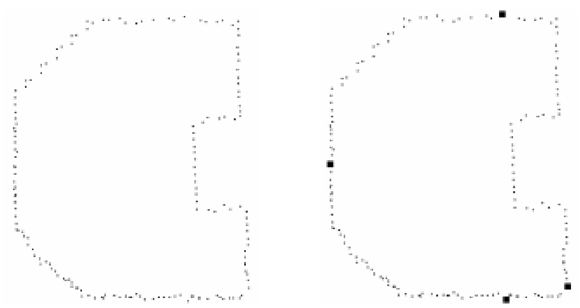
where, k is the number of first corner points

Step 3. Using only the first corner points, remove the point cp_i, if the distance from point cp_i to the line $\overline{cp_{i-1}\ cp_{i+1}}$ is shorter than H_2.

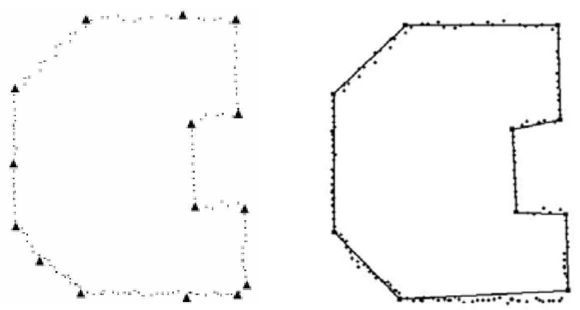
Step 4. The corner point cp_i is removed if the angle $\angle cp_i\ (= \angle cp_{i-1}\ cp_i\ cp_{i+1})$ is larger than $\alpha_{\max}$. Through this step, the points not removed based on the distance in Step 3 can be removed. This can occur for longer lines.

Step 5. In the case where the length of line $\overline{cp_i\ cp_{i+1}}$ is shorter than $L_{\min}$, compare the two angles ($\angle cp_i$, $\angle cp_{i+1}$) and remove the point with the smaller angle.

Step 6. Repeat Step3 to 5 until no more points are removed.



(a) boundary points (b) initial points



(c) first corner points (d) generalized boundary

Fig.2 The generalization process.

Figure 2 shows an example of corner points resulting from generalization; connecting the points provides approximate polygon of building boundaries. It can be seen that the corner points are well detected by our proposed algorithm based on orthogonal distance, angle, and length. As shown in Fig. 2, each line segment has a good correspondence with the edge of a real building by one-to-one relationship.

To obtain building boundaries with a higher accuracy and a more realistic shape, further processing of regularization is carried out. To do so all the boundary points as well as detected corner points are grouped. The points on the same segments, namely those between consecutive corner points are grouped into a group $G_i = \{bp_{i1}, bp_{i2}, \cdots, bp_{im}\}$. In other words, there are one-to-many correspondences between the line segment l_i and the points of group G_i .

4 Regularization

The objective of this step is to determine parametric lines and to find intersecting points of consecutive lines. This is known as regularization. To regularize the building boundaries, we first classify the buildings into three cases according to angles of corners. Here, the angles are calculated using the approximate boundaries resulting from the generalization. The three cases are as follows.

Case 1. Angles of all corners are far from being right angle.

Here, there exists no condition for the angles, so a simple least squares method can be used. The least squares solution for each line is determined mutually independently using the points corresponding to that line. For each line:

$$l_i : \; a_i x_{im} + b_i y_{im} = 1 \quad (i = 1, 2, \cdots, n) \quad (m = 1, 2, \cdots, k_i) \qquad (1)$$

where n is the number of lines and k_i is the number of points corresponding to line l_i .

Case 2. The angles of all the corners are near to being right angles.

When angles of all corners are near right angle all the lines are either parallel or perpendicular to each other. Therefore, the least squares method can be applied under the constraint that two adjacent line segments are mutually orthogonal. The following equation is used:

$$\begin{cases} a x_{im} + b y_{im} = c_i & (i = 1, 3, \cdots, n-1) \\ b x_{im} - a y_{im} = c_i & (i = 2, 4, \cdots, n) \end{cases} \qquad (2)$$

Under the constraint of orthogonal conditions, the unknown parameters of all the lines are solved at the same time to have least error regarding all the lines in a building. Because all the line segments are stored in clockwise order, i.e., adjacent line segments are not parallel but orthogonal, the upper part of Eq. (2) is applied to odd-numbered line segments and lower part of Eq.(2) is applied to even-numbered line segments.

Case 3. The angles of some of the corners are near to being right angles, and others are not.

In this case, the line segments are classified into one of three classes as they have right angles at their sides. If the angles on both sides of a line segment are far from to being a right angle, then the line segment is classified as Az_3. If at least one angle is near to being a right angle, then the line segments are classified as Az_1 or Az_2 by applying unsupervised classification method using the azimuth values of the lines

, where θ_i = the azimuth of line segment l_i, θ_{r1} = the average azimuth of Az_1, θ_{r2} = the average azimuth of Az_2, and $\tan\theta_{r1} \times \tan\theta_{r2} = -1$:

$$\begin{cases} l_i \in Az_1 & if \quad |\theta_i - \theta_{r1}| < |\theta_i - \theta_{r2}| \\ l_i \in Az_2 & if \quad |\theta_i - \theta_{r1}| > |\theta_i - \theta_{r2}| \end{cases} \tag{3}$$

For line segments classified as Az_1 or Az_2, the least squares method is conducted under the constraint of orthogonal conditions, as described in Case 2. The equations are given below, and all the unknown parameters in Eq. (4) are solved simultaneously:

$$\begin{cases} ax_{im} + by_{im} = c_i & for \;\; l \;\; of \;\; Az_1 \\ ax_{jm} - by_{jm} = d_j & for \;\; l \;\; of \;\; Az_2 \end{cases} \tag{4}$$

For line segments classified as Az_3, the simple least squares method is applied individually to each line segments, as in case 1:

$$a_k x_{kl} + b_k y_{kl} = 1 \quad (k = 1, 2, \cdots) \tag{5}$$

In any cases(Case 1, 2, 3), after the equations of the line segments are determined, i.e., all the unknown parameters are solved, then the intersection points of the line segments are found to obtain the final building boundaries. Figure 3 shows an example of a building boundary before and after our proposed regularization.

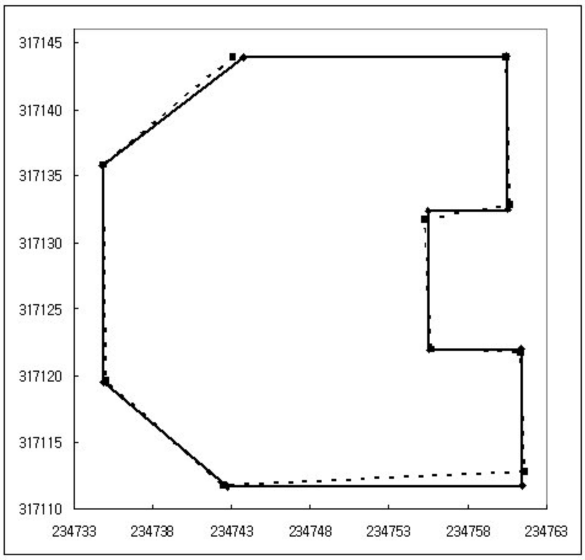
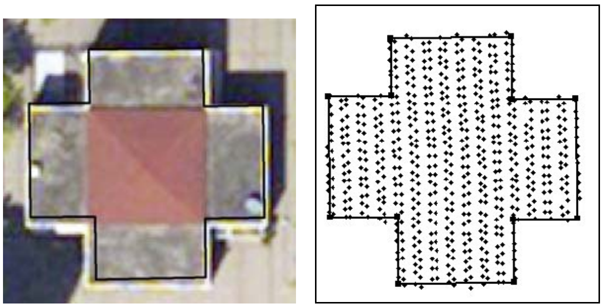


Fig.3. The result of regularization. Before regularization (dashed lines) and after regularization (solid lines)

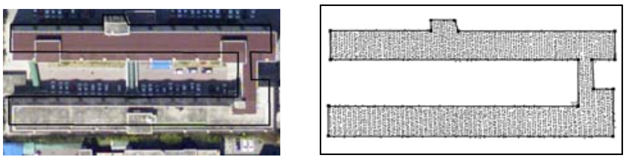
If there is at least one right angle, then all the lines that are adjacent to a right angle are adjusted simultaneously under the perpendicularity constraint using least squares method. Because all the lines adjacent to a right angle are solved simultaneously to have least error for all the points, no line segments or points are fixed as a reference, and a longer line segments has higher weights than the shorter line segments. This agrees with the fact that a longer line segment is more likely to have higher accuracy than a shorter line segments.

5 Experiments and Analysis of Results

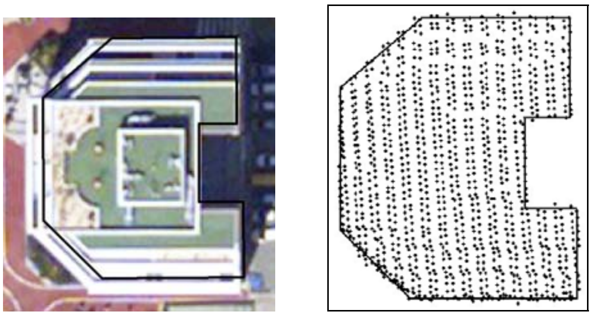
To apply our proposed approach, we selected four different building shapes that are common in many urban areas (Fig. 4). Non-rectangular edges appear in these examples. The set of algorithms from each of the steps is then applied. Thus, once the segmentation of the raw data points is complete, processes such as tracing boundary, generalization, and regularization are performed in sequence. In Fig.4, the extracted building boundaries with Lidar points in them are shown as well as corresponding aerial photos overlaid with the extracted boundaries. An accuracy assessment was carried out by comparing the resulting building boundaries with the aerial photos. From visual inspection, the extracted boundaries show a reasonably good match to the photos. However, in all the examples, some systematic shift in the extracted boundaries from the roof of the buildings in the aerial photo is found. This is because the buildings in the aerial photo are relief-displaced. An index named area overlap ratio (AOR) is used to measure the matching accuracy quantitatively. In this index, the area of overlap of two buildings – one from the proposed approach and the other from the photo – to the area of the building from the photo is calculated. Consideration the relief-displacement, the area of the building from the photo is calculated based on its ground footprint. The results are shown in Table 1. From the values given in Table 1, we can see a reasonably acceptable AOR values ranging form 93.22% to 96.75%. Therefore, our proposed approach is robust for various types of sizes of buildings. In addition, it can be seen that the concave corners are extracted well, which are usually more difficult to deal with than convex case.



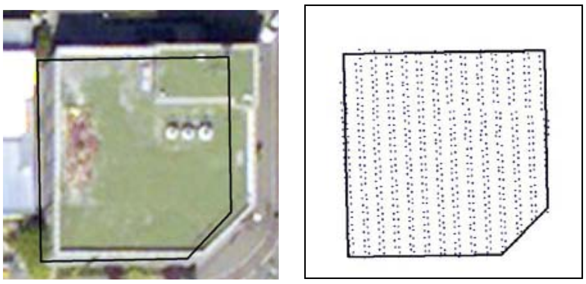
Building (a)



Building (b)



Building (c)



Building (d)

Fig.4 The final building boundaries atop the aerial photos (left), and with the original LiDAR points(right).

Table 1. Area Overlap Ratio (AOR)

	(a)	(b)	(c)	(d)
Area from photo	401.39	2523.44	748.4	574.77
Area from Lidar	378.44	2414.09	715.12	580.09
Overlapped Area	374.18	2378.78	713.14	556.09
AOL (%)	93.22	94.26	95.29	96.75

6 Conclusions

In this paper, we have proposed a new approach for obtaining building boundaries from segmented LiDAR data points, which is applicable to complex building shapes. The approach consists of three steps. In the first step, boundary tracing, the conventional convex hull algorithm was modified to applicable to non-convex buildings by restricting the search space to the current grid cell and the next cell. Then, in the second step, by taking into account the peculiarity of buildings, a generalization suitable for any building was proposed based on the orthogonal distance, angle, and the length of the building edge. The Douglas-Peucker algorithm was partially used in this step. Finally, in the third step, the regularization was carried out by suggesting different least square s equations that depends on the building types, i.e., whether it has non-rectangle edges or not.

From the results of visual and numerical tests, our proposed approach has good potential for extracting various types of building boundaries. However, it requires further experimentation to ensure that our approach is robust for any complex type of building. Therefore, more effort is needed to include various types of buildings. In particular, it would be very interesting to examine whether our approach is applicable to circular buildings.

Acknowledgments

This research is supported in part by Ministry of Construction and Transportation and the Institute of Engineering Science at Seoul National University.

References:

[1] Alharthy, A., Bethel, J., Heuristic Filtering and 3D Feature Extraction from LiDAR Data., *ISPRS Commission III*, September 9-13, 2002, Graz, Austria.

[2] Douglas, D., Peucker, T., Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature, *The Canadian Cartographer*, Vol. 10, No. 2, 197., pp. 112-122.

[3] Jenks, G. F., Geographic Logic in Line Generalization, *Cartographica*, Vol. 26, No. 1, 1989, 27-42.

[4] Lee, D. S., Shan, J., Generalization of Building Polygons Extracted from IKONOS Imagery, *ISPRS Commission IV*, July 2002, Ottawa, Canada.

[5] Lee, I., Shenk, T., 3D Perceptual Organization of Laser Altimetry Data, *International Archives of Photogrammetry and Remote Sensing*, Vol. 34, 2001, pp. 57-65.

[6] Lohmann, P., Segmentation and Filtering of Laser Scanner Digital Surface Models, *International*

Archives of Photogrammetry and Remote Sensing, 33, Amsterdam, 2001.

[7] Ma, R., DEM Generation and Building Detection from LiDAR Data, *Photogrammetric Engineering and Remote Sensing*, Vol.71, No.7, 2005, pp. 847-854.

[8]Mass, H., Vosselman, G., Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data, *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 54, No.2/3, 1999, pp. 153-163.

[9] Mayer, S., Constrained Optimization of Building Contours from High-Resolution Ortho-images, *IEEE International Conference on Image Processing*, 2001, Thessaloniki, Greece

[10] Morgan, M., Habib, A., Interpolation of LiDAR Data and Automatic Building Extraction, *ACSM-ASPRS Annual Conference Proceedings*.

[11] Rottensteiner, F., Briese, C., A New Method for Building Extraction in Urban Areas from High-Resolution LiDAR Data, *ISPRS Commission III*, September 9-13, 2002, Graz, Austria.

[12] Sampath, A., Shan, J., Urban Modeling Based on Segmentation and Regularization of Airborne LiDAR Point Clouds, *ISPRS Commission III*, July 2002, Istanbul, Turkey.

[13] Sithole, G., Filtering of Laser Altimetry Data Using a Slope Adaptive Filter, *International Archives of Photogrammetry and Remote Sensing*, 34, 3W4, 2001.

[14] Sohn, G., Dowman, I.J., Extraction of Buildings from High Resolution Satellite Data, *Proceedings of the Third International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, 10-15 June, 2001, Centro Stefano Franscini, Monte Vertia, Ascona, Switzerland.

[15] Weidner, U., Forstner, W. Towards Automatic Building Extraction from High-Resolution Digital Elevation Models, *ISPRS Journal of Photogrammetry and Remote Sensing*, 50, 1995, pp. 38-49