

Building Boundary Tracing and Regularization from Airborne Lidar Point Clouds

Aparajithan Sampath and Jie Shan

Abstract

Building boundary is necessary for the real estate industry, flood management, and homeland security applications. The extraction of building boundary is also a crucial and difficult step towards generating city models. This study presents an approach to the tracing and regularization of building boundary from raw lidar point clouds. The process consists of a sequence of four steps: separate building and non-building lidar points; segment lidar points that belong to the same building; trace building boundary points; and regularize the boundary. For separation, a slope based 1D bi-directional filter is used. The segmentation step is a region-growing approach. By modifying a convex hull formation algorithm, the building boundary points are traced and connected to form an approximate boundary. In the final step, all boundary points are included in a hierarchical least squares solution with perpendicularity constraints to determine a regularized rectilinear boundary. Our tests conclude that the uncertainty of regularized building boundary tends to be linearly proportional to the lidar point spacing. It is shown that the regularization precision is at 18 percent to 21 percent of the lidar point spacing, and the maximum offset of the determined building boundary from the original lidar points is about the same as the lidar point spacing. Limitation of lidar data resolution and errors in previous filtering processes may cause artefacts in the final regularized building boundary. This paper presents the mathematical and algorithmic formulations along with stepwise illustrations. Results from Baltimore city, Toronto city, and Purdue University campus are evaluated.

Introduction

Airborne lidar (light detection and ranging) technology provides georeferenced 3D dense point measurements over a reflective surface on the ground (Baltsavias, 1999; Wehr and Lohr, 1999). This paper discusses extracting building boundary outlines from raw lidar datasets over urban areas. As a prerequisite for many building extraction approaches, the ground points need to be separated from non-ground points, for which a number of methods have been developed. Representatives include early work by Lindenberger (1993) and Kilian *et al.* (1996) based on mathematical morphology; by Kraus and Pfeifer (1998) using least squares surface fitting; and by Axelsson (1999) and Vosselman (2000) using slope-based filters. Some recent effort focuses on the performance comparison and evaluation as reported

in Sithole and Vosselman (2004), Zhang *et al.* (2004), Shan and Sampath (2005), and Zhang and Whitman (2005), to which the readers may refer for methodological details and a comprehensive review on this topic.

Many attempts have been made on building extraction from lidar points or a digital surface model (DSM) generated from stereo images. Weidner and Förstner (1995), Brunn and Weidner (1997), and Ameri (2000) use the difference between DSM and digital terrain model (DTM) to determine the building outlines. Haala *et al.* (1998), Brenner (2000), and Vosselman and Dijkman (2001) use building plan maps, and Sohn and Dowman (2003) use Ikonos imagery to facilitate the detection and reconstruction of buildings from lidar points. Masaharu and Hasegawa (2000) segment building polygons from neighboring non-building regions, and use boundary-tracing methods to segment individual buildings. Wang and Schenk (2000) generate the triangulated irregular network (TIN) model from the lidar point clouds. Triangles are then grouped based on the orientation and position to form larger planar segments. The intersection of such planar segments results in building corners or edges. Al-Harthy and Bethel (2002) determine the building footprints by subtracting DTM from DSM obtained by initially filtering out the non-ground points. The building polygon outline is then obtained by using a rotating template to determine the angle of highest cross-correlation, which suggests the dominant directions of the building. Morgan and Habib (2002) first determine the break-lines in a raw lidar dataset and form the TIN model. Through a connected component analysis on the TIN model individual buildings are segmented. The final building boundary is formed by performing the Hough transform to the centers of the edge triangles in the TIN model. Rottensteiner and Briese (2002) use hierarchical robust interpolation (Kraus and Pfeifer, 1998) with a skew error distribution function to separate building and ground points. After applying morphological filters to the candidate building points, an initial building mask is obtained, which is then used to determine polyhedral building patches with a curvature-based segmentation process. The final individual building regions are found by a connected component analysis. For a comprehensive literature review, readers may refer to Vosselman *et al.* (2004) who present several techniques for segmenting aerial and terrestrial lidar point clouds into various classes and extracting different types of surfaces. To a similar level of extent,

Photogrammetric Engineering & Remote Sensing
Vol. 73, No. 7, July 2007, pp. 805–812.

Geomatics Engineering, School of Civil Engineering, Purdue University, 550 Stadium Mall Drive, West Lafayette, IN 47907-2051 (jshan@ecn.purdue.edu).

0099-1112/07/7307-0805/\$3.00/0
© 2007 American Society for Photogrammetry and Remote Sensing

Brenner (2005) reviews various building reconstruction techniques from images and lidar data.

Building boundary determination is a crucial and difficult (Rottensteiner and Briese, 2002) step in the building reconstruction task. As addressed earlier, some studies use available building plan maps (Haala *et al.*, 1998; Brenner, 2000; Vosselman and Dijkman, 2001), which help reduce the searching space for the estimation of the parameters of adjoining planar patches. Others, e.g., Wang and Schenk (2000), Morgan and Habib (2002) use a TIN model to determine the building boundary from lidar data. Still others (Suveg and Vosselman, 2004; Fu and Shan, 2004) use a number of rectangles to approximate and reconstruct the building boundary. In addition, Weidner (1997), Sohn and Dowman (2003), Suveg and Vosselman (2004) discuss the principle of the minimum description length to determine and regularize the building boundary. As a matter of fact, determining the boundary of a point cloud is theoretically not a trivial problem. The fundamental reason for this is the presence of concavity in the building boundary. Researchers in computational geometry have studied various approaches to attack this problem. Jarvis (1977) modifies the convex hull formation algorithm to limit the searching space to a certain neighborhood. However, this approach is not very successful in the experiments because the distribution of the used points is far from even. Edelsbrunner *et al.* (1983) propose a so-called α shape determination algorithm, where the shape of a point set is defined as the intersection of all closed discs with radii $\frac{1}{\alpha}$. This method is computationally complex, and its performance depends on the parameter α . A more recent paper by Mandal and Murthy (1997) suggests some techniques for estimating the best α to determine the boundary.

This paper is focused on building boundary tracing and regularization from raw lidar points. It consists of the following four sequential steps. First, the raw lidar points are separated to ground and non-ground two classes. Next, a moving window is used to segment individual buildings from the non-ground dataset. In the third step, a modified convex hull formation algorithm is applied to find the building boundary points and connect them to form the boundary. The final step is to regularize the building boundary by determining its parametric equations and enforcing the rectilinear constraints. A hierarchical least squares solution is developed to ensure the robustness and precision of the regularization results. Comparing with existing studies, the presented approach directly works with raw lidar points and any potential loss of information due to the gridding or rasterization process is avoided. The modified convex hull formation approach provides reliable and satisfactory building boundary without complicated shape analysis. The uneven spacing of lidar points in along and across scan directions is properly considered to achieve the best performance for building boundary tracing. By using a hierarchical least squares solution strategy, the final regularized building outline is precisely determined with quantitative quality measures and is robust to the results of the initially traced building boundary. The proposed approach is presented and evaluated both visually and numerically with three airborne lidar datasets.

Building Segmentation

As a prerequisite for building segmentation, the raw lidar data needs to be separated into two classes, ground and non-ground. This is accomplished by using the sloped

based 1D bi-directional filter proposed by Shan and Sampath (2005). In brief, the filter detects large slopes and records the local elevation along the lidar profile. Lidar points between a large positive slope and large negative slope will be labeled as non-ground points. To ensure the robustness, the filtering process is performed twice in opposite directions along the lidar profile. It is reported that 97 percent of the lidar points can be correctly separated with this approach.

In a dataset that contains only non-ground points, the task of building segmentation is to find the different clusters of points that belong to an individual building. That is, each lidar point is mapped to one building. The raw lidar points usually have a rather uniformly dense spatial distribution. In the building dataset, however, only the clusters of points that belong to one building will still have the same spatial distribution. This uniform distribution of points within one cluster and non-uniform distribution of points among clusters is used to map each point to an individual building. The solution is based on a region-growing (moving window) algorithm by successively collecting points of the same building. This algorithm consists of the following steps:

1. Start from a building point P_0 .
2. Center a window at the point and collect all the points $A = \{P_1, P_2, \dots, P_k\}$ that fall within the window.
3. Move the window center to P_1 .
4. Collect the points that fall within the window and store them in a temporary array, $T = \{tP_1, tP_2, \dots, tP_k\}$.
5. Move the window center to point P_2 . Append the newly collected points to the array T , and in this process make sure that no two points are identical.
6. Continue the process till the window has been placed over all the points in the set A .
7. Merge points in A and points in T and store them in B , i.e., $B = \{B \cup A \cup T\}$. Initially B is a null set.
8. Replace points in A with points in T such that the newly populated set A is equivalent to $\{T \setminus A\}$.
9. Go back to Step 3.
10. Stop when no new point is added to the set B .

At the end of the above steps, the set B has the points that belong to the same building. The set of points in B are removed from the original dataset and the algorithm continues to work on the rest of the points until all the points are mapped to a building. The window described in Step 2 is oriented along and perpendicular to the scan directions. The dimensions of the window are set as slightly larger than two times the point spacing, which is usually different in the across and along scan directions (Shan and Sampath, 2005). Segmented clusters containing less than a certain number of points are rejected as non-building points since they are likely vehicle or trees that are not removed from the filtering process. It should be noted that the input to this segmentation process is unstructured 3D building points. Also, the number of searches progressively reduces as more points are assigned to a building. Figure 1 shows a portion of the segmentation results, where different segmented buildings are shown with symbols of different sizes and shapes.

Boundary Tracing

Once all points of a building are found, the next step is to determine the building boundary. As pointed out earlier, many studies have been done to solve such shape determination problem for a given set of points. An early method proposed by Jarvis (1977) shows that a modified convex hull algorithm can be used to determine the shape of a set

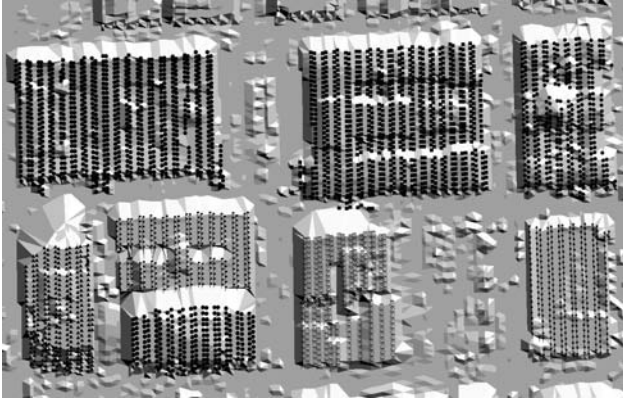


Figure 1. Segmented buildings labeled with symbols of different sizes and shapes.

of points. The modification is to restrict the searching space of a convex hull formation algorithm to a neighborhood. The study shows that the approach can yield satisfactory results if the point distribution is consistent throughout the point set. Since only the points within one building are considered, such consistency can be assumed and thus this idea is adopted and modified to trace the building boundary.

For a given set of points the convex hull is the smallest convex boundary containing all the points (de Berg *et al.*, 2000); it can be understood as a “rubber-band” wrapped around the “outside” points. As shown in Figure 2,

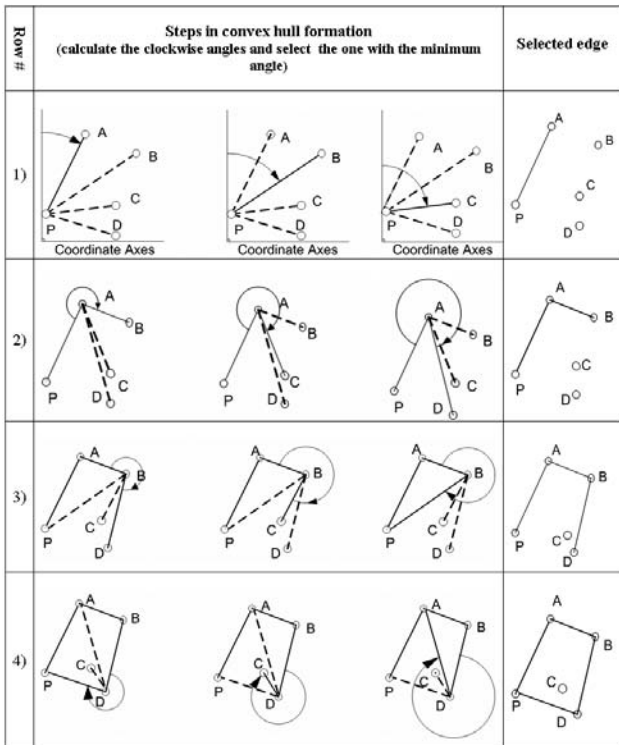


Figure 2. Convex hull formation for a set of points.

the left-most point (P) is initially selected. Next, line segments are formed between P and the rest of the points in the given set. These points are then sorted in increasing order of the clockwise angle from the vertical axis. The point that corresponds to the least angle is chosen as the next point (A). In the second step, line segments are formed between the point A and all other points. Then, the points are sorted in order of their angles between the line segment AP (the previously formed convex edge) and all other line segments. As can be seen in the second row of Figure 2, the point B corresponds to this minimum angle and is therefore chosen. The algorithm continues till it reaches the start point P . For more discussion on the convex hull problem, the reader is referred to de Berg *et al.* (2000).

Figure 3 illustrates the principle steps of the modified convex hull approach to tracing the boundary for a set of points. As shown in the first row of Figure 3, the algorithm starts with selecting the left-most point (shown by an empty circle) as the boundary point. All points (shown by gray circles) within a neighborhood (shown by a larger circle) are selected. The convex hull algorithm is then used to determine the next point on the boundary by only considering the points within this neighborhood. After that, the algorithm will proceed to this newly determined point and repeat the same procedure until the boundary is determined (fourth row in Figure 3). As can be expected, the performance of the algorithm depends on the neighborhood used in the tracing process. Since the point spacing in along and across scan directions is usually different (Shan and Sampath, 2005), a rectangular neighborhood is used, whose dimensions are slightly larger than twice of the point spacing in the along and across scan directions. In this way, only immediate adjacent points at about one ground-spacing are considered for the convex hull algorithm, such that a compact boundary is found.

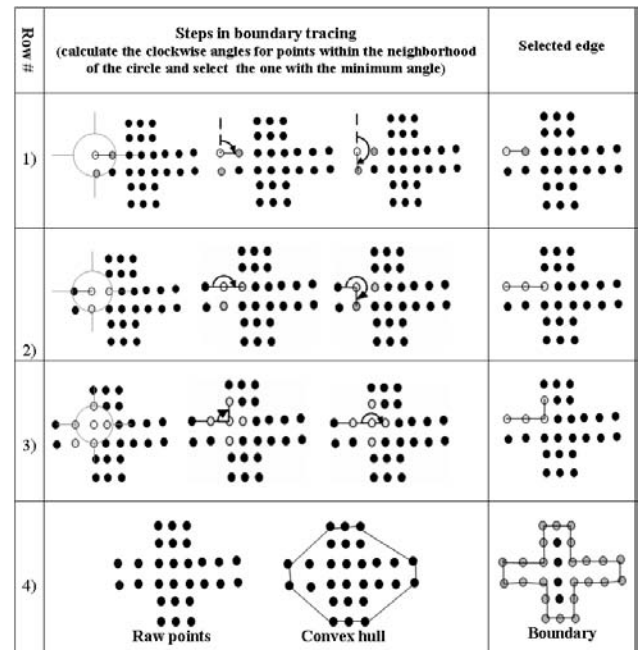


Figure 3. Modified convex hull algorithm for boundary tracing.

Based on the above discussion, the proposed boundary tracing approach is designed as follows. Let B be the set of points belonging to one building.

1. Start from point P that is necessarily a boundary point (e.g., the left-most point).
2. Select a set of points $Pts = \{P_1, P_2, \dots, P_m\} \subset B$ such that all points in Pts lie within the neighborhood of the point P .
3. Using the convex hull approach described above, determine the next boundary point P_k from $P_k \subset Pts$. The point is chosen such that the line segment $\overline{PP_k}$ does not intersect any existing line segments.

4. Choose P_k as the next current point and repeat Steps 2, 3, and 4.
5. Continue the above steps until the point P_k corresponds with the point P selected in Step 1.

Demonstrated in Figure 4 are three building boundaries obtained with this approach. The first row presents the building points and the second row shows the generated building convex hull. Clearly, it does not follow all the boundary points, nor does it represent the building shape correctly. The third row is the boundary traced by the proposed approach where the searching space of the

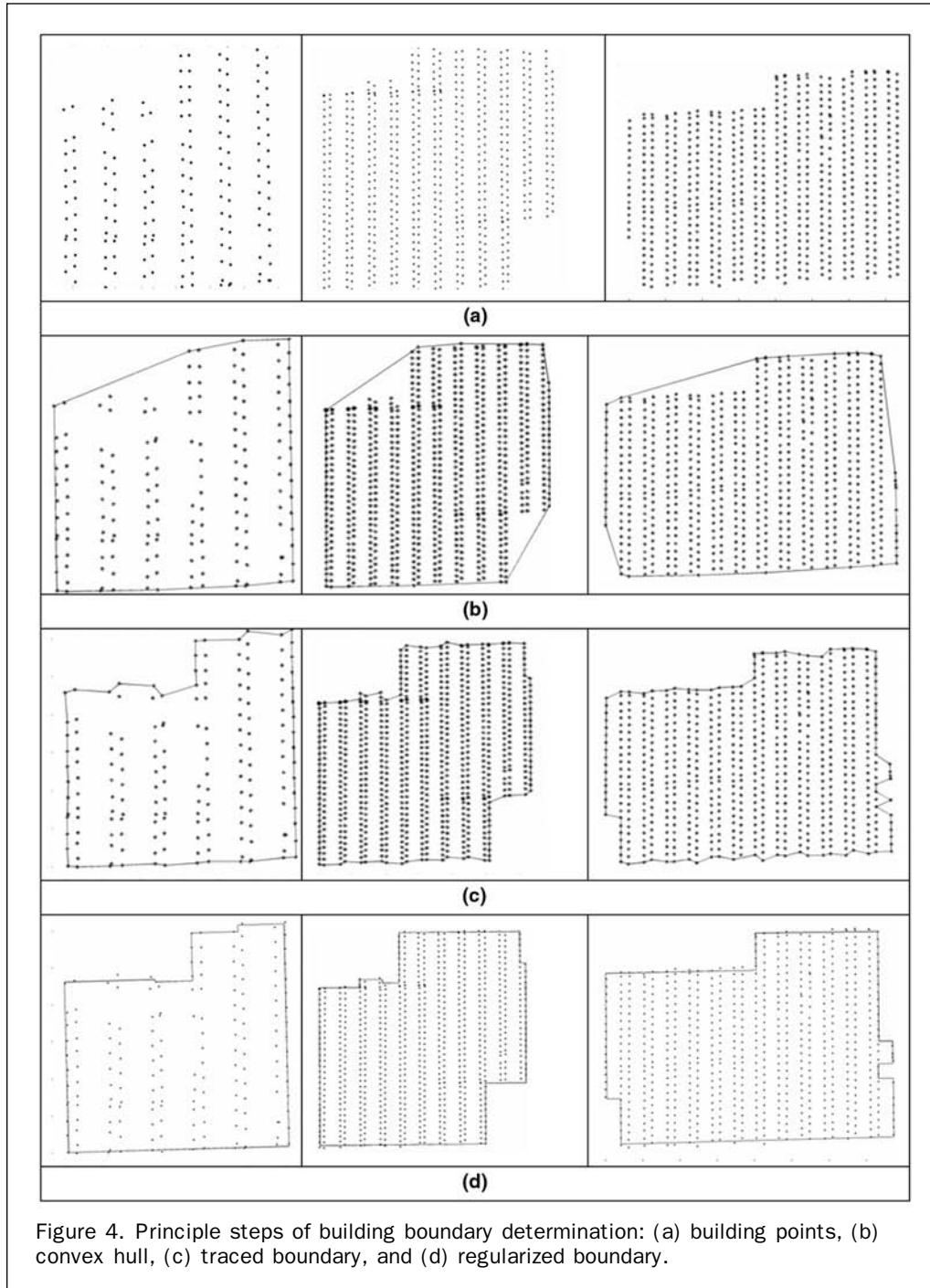


Figure 4. Principle steps of building boundary determination: (a) building points, (b) convex hull, (c) traced boundary, and (d) regularized boundary.

convex hull algorithm is restricted to a rectangular neighborhood.

Boundary Regularization

Since lidar points are randomly collected, the traced boundary (Figure 4, third row) can not be directly used as the final building boundary due to its irregular shape and possible artifacts introduced in the previous steps. Further refinement must be carried out before they can be input into a geospatial database. It is noticed that many buildings have mutually perpendicular directions. We can then use the traced boundary points to determine these directions and fit parametric lines representing the building boundary. For this regularization process, we propose the following hierarchical least squares solution.

As can be seen in Figure 4, longer edges of a building are more likely to represent its dominant directions and form the basic frame of the building boundary. The first step in regularization is therefore to extract the points that lie on longer line segments. This is done by sequentially following the boundary points and looking for positions where the slopes of two consecutive edges are significantly different. Points on consecutive edges with similar slopes are grouped to one line segment. In this way, we form a set of line segments $\{l_1, l_2, \dots, l_n\}$, from which longer (>10 meters in this study) line segments $\{L_1, L_2, \dots, L_k\}$ are then selected.

As an initial processing step, each of the selected long line segments is modeled by equation $A_i x + B_i y + 1 = 0$.

For each line segment (A_i, B_i) , the slope $M_i = -\frac{A_i}{B_i}$ is

obtained. Line segments that are parallel within a given tolerance are sorted into one group. In this way, the long line segments of a building boundary are grouped into two "horizontal" and "vertical" groups based on their slopes.

The next step is to determine the least squares solution to these long line segments, with the constraints that the slopes of these segments are either equal (parallel lines), or their product is equal to -1 (perpendicular lines) depending on whether they belong to the same category or different categories. The solutions consist of a set of parameters that describe each of the long line segments $\{L_1, L_2, \dots, L_k\}$. Specifically, we have the following formulation for the regularization problem. For each line segment, the following line equation is formed:

$$A_i x_j + B_i y_j + 1 = 0 \quad \begin{matrix} i = 1, 2, \dots, n; \\ j = j(i) = 1, 2, \dots, m_i \end{matrix} \quad (1)$$

where n is the number of line segments, and m_i is the number of points on the line segment i . Let M_u and M_v be the slopes that define the two mutually perpendicular directions of a building. For each line segment we have

$$\frac{A_i}{B_i} + M_s = 0 \quad i = 1, 2, \dots, n \quad (2)$$

where M_s is the slope for the "vertical" or "horizontal" line segment groups, and it takes either M_u or M_v . Finally, we have the following orthogonal constraint equation:

$$M_u M_v + 1 = 0. \quad (3)$$

The least squares criterion is used to solve the above equation system of Equation 1, 2 and 3. The unknowns include all the line segment parameters A_i and B_i ($i = 1, 2, \dots, n$), and the dominant slopes M_u and M_v .

The last step is to include all (long and short) line segments into the least squares solution. The slopes, M_u and M_v obtained from the previous step are used as approximate values. The slope parameters for these long line segments are given high weights in the regularization adjustment. Therefore, no explicit constraint (Equation 3) is enforced in this final step. In this way, the line segments of a building can be properly constrained to its dominant directions and have the flexibility to fit to the lidar boundary points.

Summarizing the above sequence of steps, the proposed solution is a hierarchical regularization approach. Initially, relatively long line segments are extracted from the building boundary and their least squares solution is determined, assuming that these long line segments lie on two mutually perpendicular directions. In the next step, all the line segments are included to determine the least squares solution, using the slopes of the long line segments as weighted approximations. Figure 4 (fourth row) presents the determined parametric line segments for the building boundaries.

Evaluation

Lidar data over Baltimore city (Maryland), Toronto city (Canada), and Purdue University campus (West Lafayette, Indiana) is used to evaluate the proposed approach. The data was collected with Optech™ lidar equipment, and the first returns are used for the study. The point spacing for the Baltimore data is 2.5 m and 4.0 m, respectively, for the along scan direction and the across scan direction. For the Toronto and Purdue campus datasets, the values are respectively 1.0 m and 1.5 m, and 1.0 m and 1.2 m. The point density, calculated by using 50,000 points divided by their area of extension, is 1.3 points per 10 square meters for Baltimore, 7.2 points per 10 square meters for Toronto, and 9.5 points per 10 square meters for the Purdue campus. Because of the non-uniform point spacing in different directions, we suggest to use the reciprocal of the point density as the (average) point spacing. Thus, the point spacing for Baltimore, Toronto, and Purdue are respectively 2.7 m, 1.2 m, and 1.0 m. The dimension of buildings in the datasets is at the range of several dozens of meters. Most buildings are either parallel or perpendicular to the flight direction, however, some have skew angles from several up to 45 degrees. Orthoimages are available for Baltimore city and Purdue campus, and used as independent reference for evaluation. In addition, the regularization results are also evaluated by examining the numerical quality measures obtained from the least squares adjustment.

Shown in Figures 5, 6, and 7 are the regularized buildings in Baltimore (with orthoimage), Toronto, and Purdue campus (with orthoimage), respectively. In these figures, the lidar points and regularized building boundary are overlaid atop the lidar surface model; d is the maximum distance of lidar points off the corresponding parametric line segments, and σ is the standard deviation of the least squares adjustment based regularization.

Several observations can be made based on the results in Figures 5, 6, and 7. It is first seen that almost all building edges are very well determined. The regularized boundary fits to the lidar points and reflects the building's shape. The building outline provides an authentic appearance comparing to the orthoimages and the lidar surface model. Secondly, many minor rectilinear features, e.g., the short right-angle edges labelled as **A** in a circle, are determined correctly through the regularization process. This forms the fine details in the determined building boundary, which can

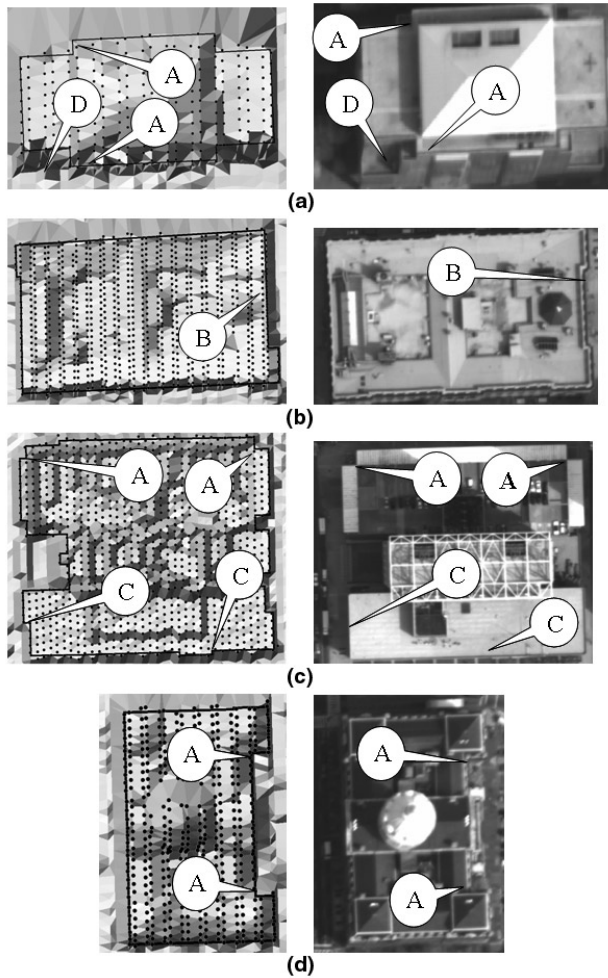


Figure 5. Regularized building, orthoimage, and quality (Baltimore): (a) $d = 2.06$ m, $\sigma = \pm 0.43$ m, (b) $d = 2.72$ m, $\sigma = \pm 0.54$ m (c) $d = 3.71$ m, $\sigma = \pm 0.64$ m, and (d) $d = 1.73$ m, $\sigma = \pm 0.36$ m.

possibly be inferred from the lidar data. The effect of lidar data resolution is our third observation. The regularized building boundary may miss details or introduce artefacts due to the limited resolution of the lidar data. As shown by the **B** labels, right-angle corners formed by short edges may not be observed in the regularized building boundary. Similarly, the **C** labels demonstrate the right-angle corners introduced as artefacts. For either of the two situations, the regularization process will produce slightly distorted and shifted building boundary segments. Finally, it is observed that very low places of a building may be identified as ground, e.g., the **D** labels in Figures 5 and 7. This in turn will cause missing parts in the final building boundary, such that the regularization result is similar to the roof print other than to the footprint.

Table 1 lists the average statistics obtained from 10 buildings, respectively, from Baltimore, Toronto, and Purdue campus datasets. The maximum distance listed in Table 1 is the average of the maximum distances of the 10 buildings, while the standard deviation is obtained from the variance average of the 10 buildings. These two measures are used to quantitatively evaluate the fitness of the regularized building

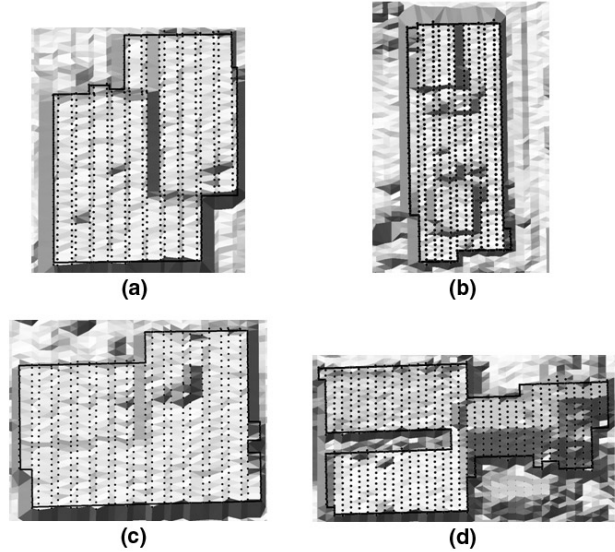


Figure 6. Regularized building and quality (Toronto): (a) $d = 0.82$ m, $\sigma = \pm 0.15$ m, (b) $d = 0.94$ m, $\sigma = \pm 0.12$ m (c) $d = 1.60$ m, $\sigma = \pm 0.23$ m, and (d) $d = 1.11$ m, $\sigma = \pm 0.23$ m.

boundary to the lidar points. Clearly, the regularization quality is dependent on the point density of the lidar data. It is interesting to note that all study sites suggest that the maximum distance of the regularized building boundary off the lidar points is about the same as the lidar point spacing: 2.4 m versus 2.7 m for Baltimore, 1.1 m versus 1.2 m for Toronto, and 1.2 m versus 1.0 m for Purdue. The standard deviations are at 18 percent to 21 percent of the lidar point spacing. Such a relationship varies little among different datasets, which suggests a trend of linear relationship between regularization uncertainty and lidar point spacing.

Conclusions

Determining building boundary from raw lidar data are solved in a four-step process: separation, segmentation, boundary tracing, and boundary regularization, among which the latter two are essential to its success. By restricting the searching space to a rectangular neighborhood, the modified convex hull algorithm can effectively trace the boundary points and form the initial good approximation of the building boundary. The hierarchical least squares solution produces satisfactory rectilinear building boundaries and is robust to boundary tracing results. The maximum offset of the determined building boundary from raw lidar points is about one lidar point-spacing. The regularization uncertainty is in average 18 percent to 21 percent of lidar point spacing, and such a relationship varies little to the lidar point spacing, which suggests the uncertainty of the regularized building boundary tends to be linearly proportional to the lidar point spacing. Limitation of lidar data resolution and errors in previous filtering process may cause artefacts in the final regularized building boundary. Future effort will be made to extend the presented approach to handle buildings with multiple (>2) and non-perpendicular, dominant directions and buildings with complex structures, such as inner yards and non-linear boundaries.

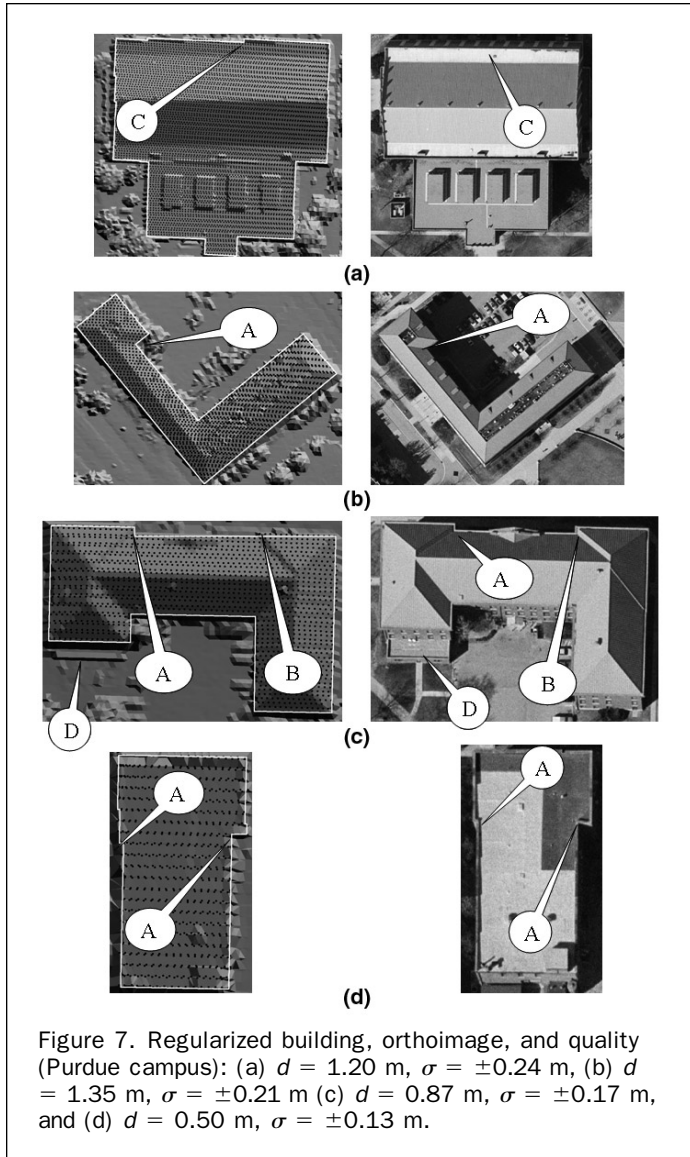


Figure 7. Regularized building, orthoimage, and quality (Purdue campus): (a) $d = 1.20$ m, $\sigma = \pm 0.24$ m, (b) $d = 1.35$ m, $\sigma = \pm 0.21$ m (c) $d = 0.87$ m, $\sigma = \pm 0.17$ m, and (d) $d = 0.50$ m, $\sigma = \pm 0.13$ m.

TABLE 1. QUALITY OF BUILDING BOUNDARY REGULARIZATION VERSUS POINT DENSITY (AVERAGE d IS THE MEAN OF THE MAXIMUM DISTANCES BETWEEN A LIDAR POINT AND ITS CORRESPONDING LINE SEGMENT. σ IS THE STANDARD DEVIATION OF THE LEAST SQUARES ADJUSTMENT IN REGULARIZATION. THESE STATISTICS ARE BASED ON 10 BUILDINGS IN EACH DATASET.)

	Baltimore	Toronto	Purdue
Average point spacing (m)	2.7	1.2	1.0
Average d (m)	2.37	1.14	1.17
Std dev. σ (m)	± 0.48	± 0.21	± 0.21

References

Al-Harthy, A., and J. Bethel, 2002. Heuristic filtering and 3D feature extraction, *Proceedings of ISPRS Commission III Symposium*, Graz, Austria, unpaginated CD-ROM.

Ameri, B., 2000. Automatic recognition and 3D reconstruction of buildings from digital imagery, *Deutsche Geodaetische Kommission*, Series C, No. 526.

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

Baltsavias, E., 1999. A comparison between photogrammetry and laser scanning, *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3):83-94.

Brenner, C., 2000. Dreidimensionale Gebäuderekonstruktion aus digitalen Oberflächenmodellen und Grundrissen, *Deutsche Geodaetische Kommission*, Series C, No. 530.

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

Brunn, A., and U. Weidner, 1997. Extracting buildings from digital surface models, *International Archives of Photogrammetry and Remote Sensing*, 32(3-4):27-34.

de Berg, M., O. Schwarzkopf, M. van Kreveld, and M. Overmars, 2000. *Computational Geometry: Algorithms and Applications*, 2nd edition, Springer-Verlag, 367 p.

Edelsbrunner, H., D. Kirkpatrick, and R. Seidel, 1983. On the shapes of a set of points in the plane, *IEEE Transactions on Information Theory*, IT29(4):551-559.

Fu, C.S., and J. Shan, 2004. 3-D building reconstruction from unstructured distinct points, *International Archives of Photogrammetry and Remote Sensing*, Vol. 35, Part B3, unpaginated CD-ROM.

Haala, N., C. Brenner, and K.-H. Anders, 1998. Urban GIS from laser altimeter and 2D map data, *International Archives of Photogrammetry and Remote Sensing*, ISPRS Commission III Symposium on Object Recognition and Scene Classification from Multispectral and Multisensor Pixels, (T. Schenk and A. Habib, editors), Columbus, Ohio 32(3/1):339-346.

Jarvis, R.A., 1977. Computing the shape hull of points in the plane, *Proceedings of IEEE Computer Society Conference Pattern Recognition and Image Processing*, pp. 231-241.

Kilian, J., N. Haala, and M. English, 1996. Capture and evaluation of airborne laser scanner data, *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXI, Part B3, pp. 383-388.

Kraus, K., and N. Pfeifer, 1998. Determination of terrain models in wooded areas with airborne laser scanner data, *ISPRS Journal of Photogrammetry and Remote Sensing*, 53(4):193-203.

Lindenberger, J., 1993. Laser-profilmessungen zur topographischen Geländeaufnahme, *Deutsche Geodaetische Kommission*, Series C, No. 400.

Mandal D.P., and C.A. Murthy, 1997. Selection of alpha for alpha-hull in \mathbb{R}^2 , *Pattern Recognition*, 30(10):1759-1767.

Masaharu, H., and H. Hasegawa, 2000. Three dimensional city modeling from laser scanner data by extracting building polygons using region segmentation method, *International Archives of Photogrammetry and Remote Sensing*, Vol. 33, Part B3, Amsterdam, The Netherlands, unpaginated CD-ROM.

Morgan, M., and A. Habib, 2002. Interpolation of lidar data and automatic building extraction, *ACSM-ASPRS Annual Conference Proceedings*, unpaginated CD-ROM.

Rottensteiner, F., and C. Briese, 2002. A new method for building extraction in urban areas from high-resolution lidar data, *Proceedings of the ISPRS Commission III Symposium*, Graz, Austria, unpaginated CD-ROM.

Shan, J., and A. Sampath, 2005. Urban DEM generation from raw lidar data: A labeling algorithm and its performance, *Photogrammetric Engineering & Remote Sensing*, 71(2):217-226.

Sithole, G., and G. Vosselman, 2004. Experimental comparison of filter algorithms for bare earth extraction from airborne laser scanning point clouds, *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(1-2):85-101.

Sohn, G., and I. Dowman, 2003. Building extraction using LiDAR DEMs and IKONOS images, *International Archives of Photogrammetry and Remote Sensing*, WG III/3 Workshop on 3-D Reconstruction from Airborne Laserscanner and InSAR Data, Dresden, Germany, 08-10 October, Vol. 34, 3/W13, unpaginated CD-ROM.

Suveg, I., and G. Vosselman, 2004. Reconstruction of 3D building models from aerial images and maps, *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3-4):202-224.

Vosselman, G., 2000. Slope based filtering of laser altimetry data, *International Archives of Photogrammetry and Remote Sensing*, Vol. 33, Part B3/2, Amsterdam, the Netherlands, pp. 935-942.

- Vosselman, G., and S. Dijkman, 2001. 3D building model reconstruction from point clouds and ground plans, *International Archives of Photogrammetry and Remote Sensing*, 34(3W4):37–43.
- Vosselman, G., B.G.H. Gorte, G. Sithole, and T. Rabbani, 2004. Recognising structure in laser scanner point clouds, *International Archives of Photogrammetry and Remote Sensing*, 46(8/W2):33–38.
- Wang, Z., and T. Schenk, 2000. Building extraction and reconstruction from lidar data, *International Archives of Photogrammetry and Remote Sensing*, Vol. 33, Part B3. Amsterdam, The Netherlands, unpaginated CD-ROM.
- Weidner, U., and W. Förstner, 1995. Towards automatic building extraction from high-resolution digital elevation models, *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):38–49.
- Weidner, U., 1997. *Gebäudeerfassung aus digitalen Oberflächenmodellen*, Ph.D. Thesis, Institute of Photogrammetry, Bonn University, DGK-C 474.
- Wehr, A., and U. Lohr, 1999. Airborne laser scanning – An introduction and overview, *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2–3):68–82.
- Zhang, Y., C.V. Tao, and J.B. Mercer, 2004. An initial study on automatic reconstruction of ground DEMs from airborne IfSAR DSMs, *Photogrammetric Engineering & Remote Sensing*, 70(4):427–438.
- Zhang K., and D. Whitman, 2005. Comparison of three algorithms for filtering airborne lidar data, *Photogrammetric Engineering & Remote Sensing*, 71(3):313–324.