

Using a Binary Space Partitioning Tree for Reconstructing Polyhedral Building Models from Airborne Lidar Data

Gunho Sohn, Xianfeng Huang, and Vincent Tao

Abstract

During the past several years, point density covering topographic objects with airborne lidar (Light Detection And Ranging) technology has been greatly improved. This achievement provides an improved ability for reconstructing more complicated building roof structures; more specifically, those comprising various model primitives horizontally and/or vertically. However, the technology for automatically reconstructing such a complicated structure is thus far poorly understood and is currently based on employing a limited number of pre-specified building primitives. This paper addresses this limitation by introducing a new technique of modeling 3D building objects using a data-driven approach whereby densely collecting low-level modeling cues from lidar data are used in the modeling process. The core of the proposed method is to globally reconstruct geometric topology between adjacent linear features by adopting a BSP (Binary Space Partitioning) tree. The proposed algorithm consists of four steps: (a) detecting individual buildings from lidar data, (b) clustering laser points by height and planar similarity, (c) extracting rectilinear lines, and (d) planar partitioning and merging for the generation of polyhedral models. This paper demonstrates the efficacy of the algorithm for creating complex models of building rooftops in 3D space from airborne lidar data.

Introduction

Today, there are increasing demands for rapid and timely compilation of three-dimensional building models from remotely sensed data. Accurate acquisition and frequent up-dating of such models becomes more important source of information for decision making in support of numerous applications, including geospatial database compilation, urban planning, environmental study, and military training (Ameri, 2000). Traditionally, three-dimensional compilation of urban features has been manually conducted under guidance of a human operator using a Digital Photogrammetry Workstation (DPW). Automation of such resource-intensive tasks has been

a major focus of Photogrammetry and Remote Sensing for many years.

In recent years, topographic airborne lidar (Light Detection and Ranging) has been rapidly adopted as an active remote sensing system that uses near-infrared laser pulses (typically 1 to 1.5 μm) to illuminate man-made or natural features on the terrain. The up-to-date lidar system can collect elevation data at a vertical accuracy of 15 cm, at a rate of higher than 100,000 pulses per second. This ability allows the system to produce a dense array of highly accurate and precise three dimensional elevation models, which is a useful property for automating the sophisticated tasks involved in building reconstruction. This paper focuses on the issue of automated construction of 3D building models from lidar data.

It is well understood that a general solution to the building reconstruction system entails the collection of modeling cues (e.g., lines, corners, and planes), which represent the major components of building structure. By correctly grouping those cues, geometric topology between adjacent cues, describing a realistic roof shape, can be created. A significant bottleneck hindering the reconstruction process is caused by the fact that extraction of modeling cues is always disturbed by noise inherited from imaging sensors and objects. The most disadvantageous feature of lidar is characterized by irregular data acquisition, which often makes extraction of modeling cues difficult. As shown in Figure 1, the salient boundaries comprising building roof structures, which are easily recognizable in the optical imagery, are often distorted due to a variety of factors, most notably: scanning pattern, point spacing, surface material properties, and object complexity. For this reason, 3D building reconstruction systems have performed most effectively by constraining the knowledge of building geometry either explicitly (model-driven reconstruction) or implicitly (data-driven reconstruction) in order to recover incomplete modeling cues.

The model-driven approach pre-specifies particular types of building models so that geometric relations (topology) across modeling cues are provided. By fitting the model to observed data, the model parameters are determined. A good example of model-driven reconstruction was presented by Maas and Vosselman (1999), who were able to determine the parameters of a standard gable

Gunho Sohn and Vincent Tao are with the GeoICT Lab, Department of Earth and Space Science and Engineering, York University, 4700 Keele St., Toronto, ON, Canada, M3J 1P3 (gsohn@yorku.ca).

Xianfeng Huang is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, 129 Luoyu Road, Wuhan, China.

Photogrammetric Engineering & Remote Sensing
Vol 74 No. 11, November 2008, pp. 1425–1438.

0099-1112/08/7411-1425/\$3.00/0
© 2008 American Society for Photogrammetry
and Remote Sensing

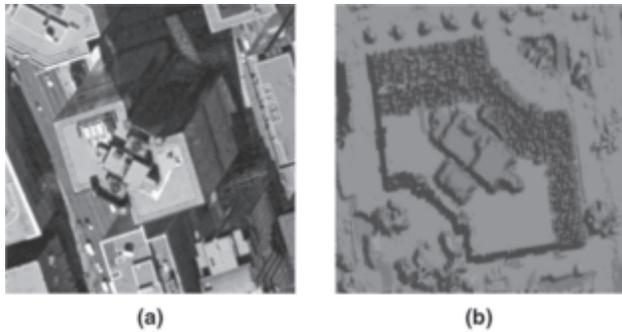


Figure 1. Lidar data acquisition property: the same building acquired from (a) an airborne optical sensor, and (b) lidar (see the lidar acquisition errors over the building in the center of the image).

roof with small dorms through the analysis of invariant moments from lidar points. In order to cope with more complicated models, a number of different building models are hypothesized and tested within partitioned rectangles of existing ground planes, and a full description of the roof model is generated by integrating the verified parametric models (Brenner and Haala, 1998; Vosselman and Dijkman, 2001).

The data-driven method reconstructs the boundary of polyhedral building models by grouping extracted lines, corners and planes with minimal use of prior knowledge of the generic building model. As the modeling cues, plane clusters are obtained by the Hough transformation (Vosselman, 1999); RANSAC (Brenner, 2000); orthogonal projection of point clouds (Schwalbe *et al.*, 2005); and region growing (Rottensteiner *et al.*, 2005). Also, linear features are extracted by intersecting planes detected and the analysis of height discontinuity (Vosselman, 1999); approximating the boundary of planes (Alharthy and Bethel, 2004). The Gestalt laws (proximity, symmetry, parallelism, and regularity) are often used for bridging the gaps between extracted features in order to reconstruct the model topology (Hu, 2003) or to snap adjacent planes (Alharthy and Bethel, 2004). In order to impose geometric regularity on reconstructed models, additional constraints are used in which: (a) a set of rules for intersecting adjacent planes are pre-determined (Hofman, 2004); (b) the roof structure is restricted to the dominant building orientation (Vosselman, 1999; Schwalbe *et al.*, 2005); or (c) orientations of building outlines are derived from an existing ground plan (Brenner, 2000). A global optimality of shape regularity is achieved by the Minimum Description Length (MDL) principle (Weidner and Förstner, 1995).

Many buildings in modern cities exhibit a very high degree of shape complexity, being comprised of a mixture of various building primitives, with multiple stories and many roof super-structures (chimneys, air vents, water tanks, roof fences, etc.). Under these circumstances, particular types of pre-defined models become problematic to accommodate high variations in building shapes (Brenner, 2005). On the other hand, strong constraints in analyzing topological relations and grouping adjacent model cues, which are commonly used in data-driven reconstruction, may degrade the robustness of reconstruction systems if the fragmentary level of cue extraction becomes higher and roof super structures are located around the border of adjacent roof primitives. Using an existing ground plan to sub-divide

a building region of interest into a set of rectangles can greatly reduce the complexity of the building reconstruction process (Brenner 2000; Vosselman and Suveg 2001). However, as building complexity becomes higher, pre-assumptions of coherent relations between roof structures and the ground plan may be invalid

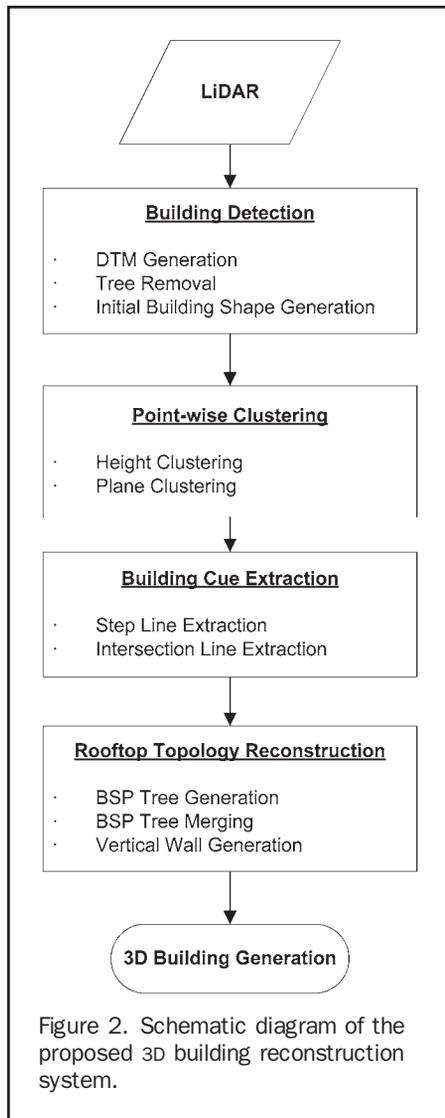
The main objective of this paper is to present a new partitioning framework, where data-driven linear features optimally subdivide a building object into piecewise plane clusters with no use of existing ground plans. Thus, 3D roof structures are reconstructed by collecting plane clusters so that fragmented linear cues can be implicitly recovered. In the following section, the major components comprising the suggested building reconstruction system are explained. Afterwards, experimental results from topographic lidar data containing various types of building objects are discussed. Finally, we draw conclusions with recommendations for future work.

Proposed Method

Overview

The boundary of a building object with planar facets is decomposed into collections of more primitive elements (i.e., modeling cues) such as facets, edges, and vertices. A generic model of 3D buildings can be represented based on a combined topological and geometric description of the building object. Here, the topology of building models is obtained by recovering a set of relations that indicate how the faces, edges, and vertices are connected to each other, while the geometry can be specified by estimating appropriate geometric parameters (perimeter, intersecting angle, area, orientation, location, etc.) of modeling cues. The purpose of our research is to automatically model building objects by recovering both topological and geometrical relations across modeling cues, primarily relying on the information driven from lidar data only, but with minimal use of *a priori* knowledge on targeted building shapes. This data driven approach is more amenable to describing variants of building shapes than is the use of pre-specified model primitives. However, a scientific challenge in the data-driven technique may lie in a reliable grouping of modeling cues that are fragmented with unpredictable order. This fragmentation can be due to many reasons, including the noise inherited from the sensor, disadvantageous backscattering properties and object complexity. In order to resolve this problem, we suggest a unique framework for globally grouping incomplete features extracted from a given data space that yields successively finer approximations of the building object.

The method consists of four steps: (a) building detection, (b) point-wise clustering, (c) linear cue extraction, and (d) building rooftop reconstruction. The functional elements of the proposed building reconstruction system are schematically illustrated in Figure 2. The main purpose of the building detection process are to classify the lidar data into building and non-building points, and to provide coarse building models for individual building objects by bounding each isolated building region with a rectangle for the subsequent reconstruction processes. Two interesting low-level features, including plane clusters and linear features, are extracted in order to serve as modeling cues to delineate rooftop structures. The classified building points are clustered by the similarity criterion of either height (height cluster) or planar property (planar cluster). An edge is a significant local change in lidar data, usually associated with a discontinuity in either height (step line) or principal curvature (intersection line). The step lines are obtained by approximating the boundary shared with adjacent height clusters. On the other hand, the intersection of adjacent



planar clusters with different slopes produces intersection lines. As the coarse building shape (rectangle) is hierarchically sub-divided by the extracted linear features, convex polygons with more fine scales are produced. Finally, a geometric topology between adjacent polygons (i.e., planes) is reconstructed when all partitioned polygons are verified as building planes. This topology reconstruction is implemented by optimally controlling the generation of the Binary Space Partitioning (BSP) tree.

Extracting Initial Building Models

Our building reconstruction system starts to extract coarse building models from lidar data, which bound individual building regions by simple rectangles. These initial building models will be later partitioned to finer models that become closer to real building shapes. An important element to create the initial building models is to reliably isolate laser points that only belong to building objects from the ones located over non-building features. Rather than directly detecting building points (Filin, 2002), we followed a hierarchical focusing strategy (Baillard, 1999) where the targeted foreground objects (i.e., buildings) are detected by removing the background features including terrain and trees. The first step was to apply a model-based filtering algorithm to airborne lidar data (Sohn and Dowman, 2008),

in order to automatically identify laser points hit on the ground only. This technique was developed, in particular to allow a lidar filter to be self-adaptive to various landforms of different slopes. This terrain filter employed a tetrahedral model to generally represent a terrain surface with single slope (homogeneous terrain). A model-fitness between lidar points and the tetrahedral terrain model was measured in a framework of Minimum Description Length (MDL). In a coarse-to-fine scheme, this model-fitness scores triggered to recursively fragment a lidar DEM convolved with heterogeneous terrain slopes into piecewise homogeneous sub-regions where underlying terrain can be well characterized by the terrain model. Once these homogeneous terrains were obtained, the separation of non-terrain points from the ground can be simply made by a height threshold, as terrain characteristics over sub-regions are uniformly regularized. Then, the building points were retained by removing points over the tree objects from non-terrain points. In order to eliminate the tree points, (a) laser points showing large height difference between first and last returns were first removed, and (b) the connected component analysis was then applied to spatially clustered non-terrain points for investigating average height, boundary regularity, and surface smoothness. A non-point cluster is removed as the tree object if the aforementioned criteria are less than thresholds. After removing tree features from the non-terrain ones, individual buildings comprising only building points are bounded by the rectangles which will be feedback as initial building models to the following building reconstruction procedures.

Height Clustering

Since we aim to reconstruct 3D buildings with a mixture of multiple flat and sloped planes containing rooftop furniture, extracting modeling cues directly from an entire building may result in difficulties due to such a high degree of shape complexity. In order to reduce this complexity, lidar points collected for individual buildings are first decomposed into a set of clusters based on height similarity. A maximum height deviation for each point is measured from its neighboring points in a triangulated irregular network (TIN), which contributes to producing a height histogram of a certain bin size. After applying a simple moving box filter to the height histogram, a maximum height peak, δ_h , is automatically found from the histogram, within a height range between 1 m to 3 m. Similarly to a conventional region growing algorithm, lidar points are grouped in one height cluster if a height discrepancy between a point and those connected to it in a TIN is less than δ_h . As a consequence, the data domain R of lidar points P located over a building rooftop are divided into a set of height clusters $\{R_i; i = 1, \dots, N\}$ so that $R = \cup_{i=1}^N R_i$, $R_i \cap R_j = \phi$ if $i \neq j$, and P satisfies a homogeneity criterion on each R_i . Note that one height cluster may include several different planes or corresponds to only one plane if it is isolated from neighboring building parts in terms of height difference (Figure 3a). During the process, all of the cue extraction algorithms are independently applied to each height cluster (local process), but a building reconstruction based on locally collected cues will be conducted regardless of this height clustering result over the entire building region (global process).

Plane Clustering

This section describes a plane clustering process that segments lidar points into a set of clusters which uniquely represent planar surfaces. In current research, these planar cues play an important role in either extracting intersection lines or in optimally partitioning the data space, thereby reconstructing the rooftop topology across the extracted

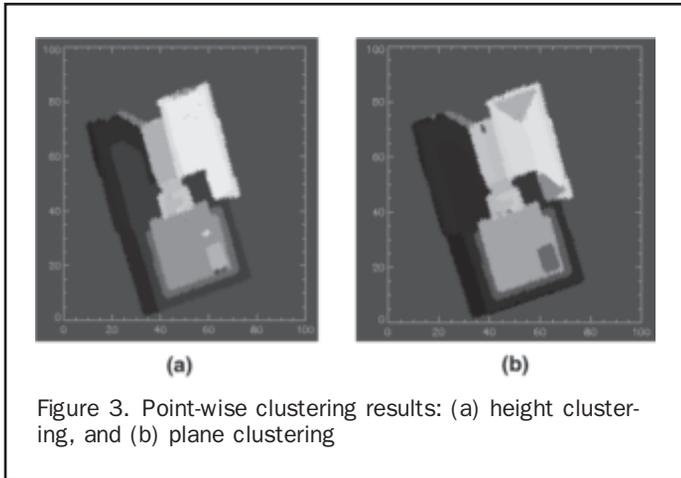


Figure 3. Point-wise clustering results: (a) height clustering, and (b) plane clustering

linear features. Thus, obtaining a reliable planar clustering result from lidar points is important. The implemented clustering process follows a hierarchical voting process in the parameter space discussed by Vosselman *et al.* (2004). In general, a three-dimensional plane passing through a point (x,y,z) in the data space can be represented by:

$$x \cos \alpha + y \cos \beta + z \cos \gamma = \rho \quad (1)$$

where (α, β, γ) is the plane normal vector angles measured between the plane and the x -, y - and z -axis, respectively, satisfying $\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma = 1$, and ρ is the distance of the plane from the origin of the coordinate system. The voting process quantizes the plane parameter $(\alpha, \beta, \gamma, \rho)$ in a discrete parameter space with certain bin sizes. The position of a particular bin in the parameter space uniquely represents a plane in the data space. The plane parameter locally estimated from the data space votes for a corresponding bin in the parameter space. Thus, by searching maximum peaks in the parameter space, the dominant planes passing through laser points of interest can be found (Vosselman *et al.*, 2004)

Rather than clustering the four plane parameters simultaneously, two separate parameter spaces for clustering (α, β, γ) , and ρ , respectively, are used in order to reduce the dimensions of the parameter space. The normal vector angles (α, β, γ) are computed for all the points using a conventional least squares method. An error of this local planar parameter estimation is measured by an orthogonal residual between the point and the plane estimated. Only for those points with an estimation error of less than 1 m, the normal vector angles are mapped in the parameter space with one unit. The maximum peak of (α, β, γ) is determined in the parameter space if the voting score is larger than a certain threshold. We collect neighboring points from the maximum peak, by which the average values of planar parameters can be determined from neighboring points, including maximum peak. The remaining parameter ρ for all points with a normal vector similar to the maximum is calculated by Equation 1 and is mapped to a one-dimensional parameter space. The maximum in this space determines the orthogonal distance of the plane from the origin. This process continues until all of the maximum peaks of the plane parameters can be found.

The previously mentioned plane clustering algorithm works on each height cluster independently. After assigning the plane parameters to the entire building, a plane adjacency graph is created for analyzing the connectivity of adjacent planes. The connected planes are merged if the

plane parameters $(\alpha, \beta, \gamma, \rho)$ are almost the same and are recomputed using all of the points of the merged plane. This plane merging process continues until no similar adjacent planes are found. Figure 3b shows a plane clustering result obtained when the presented method was applied to Figure 3a.

Step Line Extraction

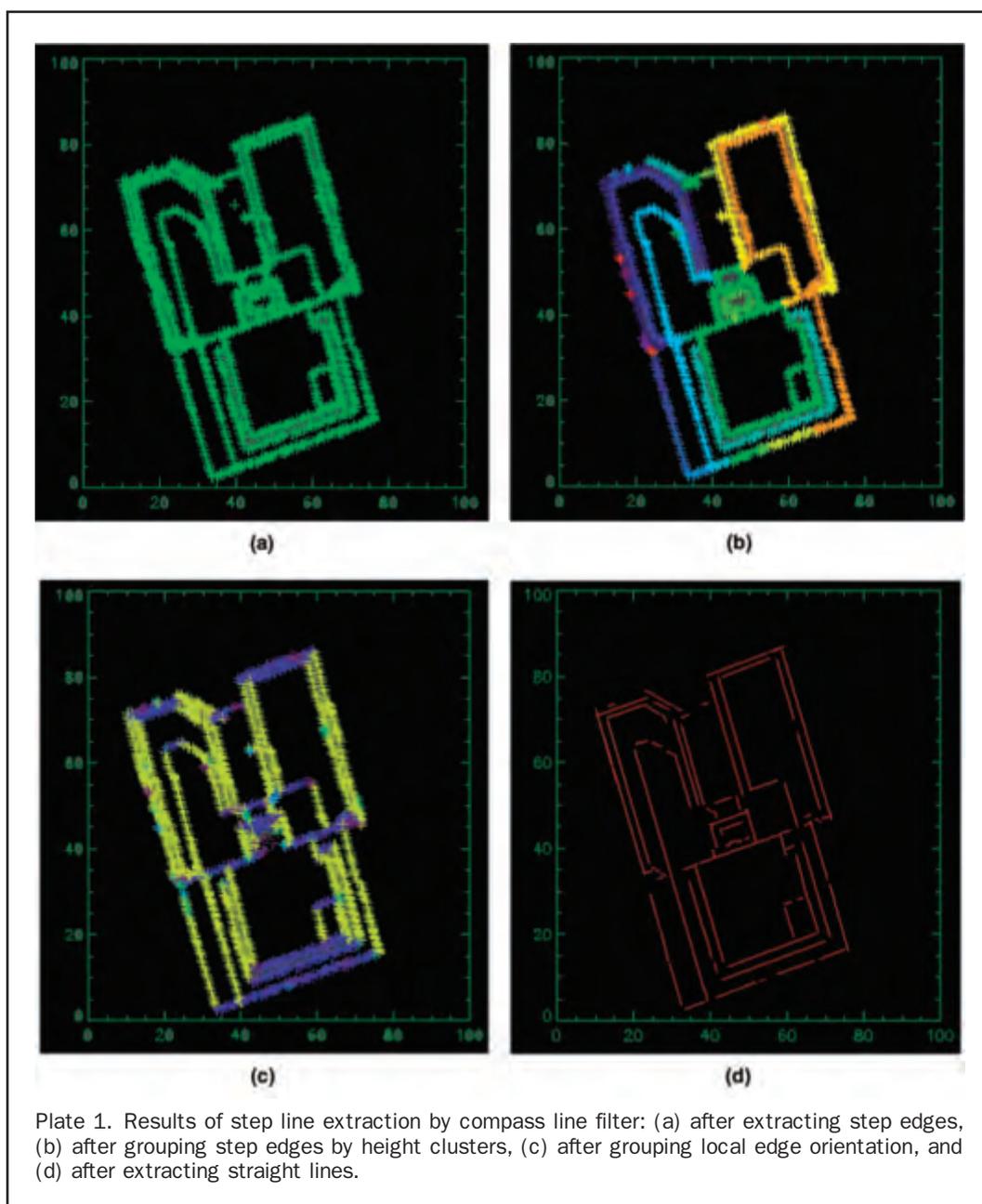
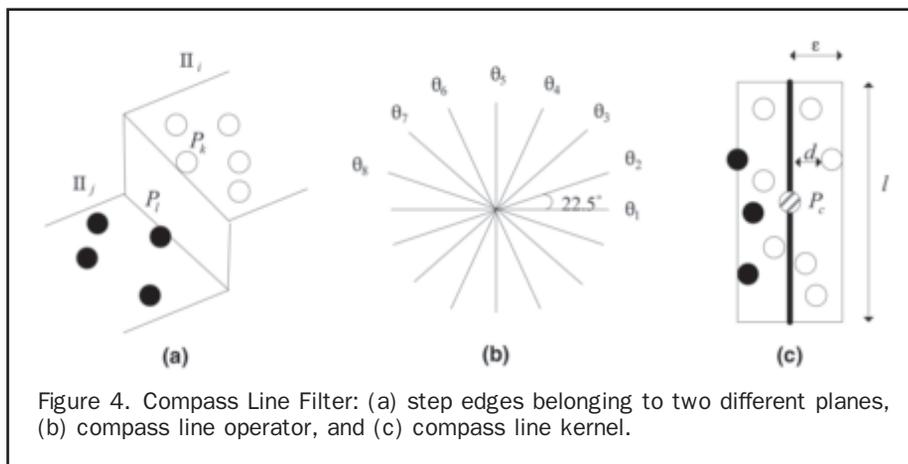
A step line is produced where the height continuity changes abruptly over rooftop surfaces. The line can be found around the building boundary or the rooftops of multiple story buildings where two planes are adjacent to each other, but with a large height discontinuity. As discussed by Vosselman (1999), compared to the optical sensors, extracting the step line by detecting the height discontinuity from lidar is not a trivial job due to the nature of discrete and irregular data acquisition. He suggested a sequential algorithm for extracting step lines from lidar data. After the connected component of the points on all roof surfaces is determined, the contour of this component is then approximated by step lines that are either parallel or perpendicular to the main building orientation. The method showed a successful result for reconstructing a row of houses with small sheds. However, it is limited to delineating only two directions, which are determined depending on the main building orientation. Obviously step lines with more than two directions are required in order to reconstruct complex buildings.

In the current experiment, we develop a step line extractor, called a Compass Line Filter (CLF), for extracting straight lines from irregularly distributed lidar points. The process starts to collect step edges, which are points showing abrupt height discontinuities relative to their neighborhoods in the TIN. Since the data R is already segmented into N height clusters $\{R_i\}_{i=1}^N$, the step edges can be simply collected by tracing all the boundaries where adjacent height clusters with different IDs share. However, as shown in Figure 4a, this process produces double step edges on the common border of two adjacent planes, (e.g., recognize P_k and P_l of planes Π_i and Π_j as step edges in Figure 4a). Since the proposed algorithm generates step lines by approximating the step edges contour, obtaining a thinner contour yields better approximation result. Thus, a thinning process is considered to a point P_c as the step edge if its neighboring points P_n belongs to more than one height cluster, and P_c is the member of the highest plane. This relation can be described by a signed height difference function, $H(P_c, P_n) = P_c - P_n$ as follows:

$$\left| \max_{\forall P_n} H(P_c, P_n) \right| > \left| \min_{\forall P_n} H(P_c, P_n) \right| \quad (2)$$

Once all the step edges are extracted (Plate 1a) and grouped by each height cluster (Plate 1b), the next step is to determine the local edge orientation for each step edge using a compass line filter. The local edge orientation provides directional information of the change of height discontinuity at a particular point. However, due to the nature of irregular data sampling, implementing a first-order of gradient operator to detecting height discontinuity from lidar is not as trivial as is using optical imagery with fixed pixel size.

We modified Kirsch's compass kernels, which can be applied to irregularly distributed point space. Instead of employing the gradient kernel with fixed size and gradient coefficients, we employ a compass line operator shown in Figure 4b) that has the whole set of eight filtering lines with different slopes $\{\theta_i: i = 1, \dots, 8\}$, each of which, as illustrated in Figure 4c, is equally separated in steps of 22.5° (the first compass line corresponds to the horizontal line). Each line has two virtual boxes (line kernel) where length l is the same as the one of line and width corresponding to ϵ .



Suppose that a height jump edge P_c belonging to a height cluster R_i is convolved with the whole set of eight line kernels. The local edge gradient is measured by the sum of distances d between the compass line θ_i and N kernel member points that are located within the line kernel K_i and belonging to R_i . A final local edge orientation θ^* is determined with the orientation of the kernel that yields the maximum local edge gradient. The values for the output orientation lie between one and eight, depending on which of the eight kernels had produced the maximum response. The computation of this local orientation for height jump edges is described by:

$$G(\theta_i) = \sum_{k=1}^N \frac{2}{(1 + e^{\varepsilon \times d_k})}, \theta^* = \arg \max_{\theta_i} (G(\theta_i)), i = 1, \dots, 8. \quad (3)$$

Once the local edge orientation is determined by the compass line operator, step edges with similar local edge orientation, belonging to the same height cluster, are grouped (Plate 1c). The local edge orientation angle is explicitly assigned by a pre-specified compass line filter. Finally, the real orientations for θ_i are newly determined by the well-known Hough transformation (Plate 1d).

Intersection Line Extraction

Once all of the planes are detected by the plane clustering process for each height cluster, the intersection lines are obtained by intersecting two adjacent planes. Thus, accuracy of the intersection line computation is subjective to the plane clustering result. The major problem of the plane clustering method presented in the previous section is that the process follows a *winner-take-it-all* approach, which does not take into account of a retro-validation in the hierarchical voting scheme. Suppose that we try to extract intersection lines over a simulated hip roof with certain amount of white noise (Figure 5a). If a plane Π_i is detected with higher scores, more numbers of points (white dots in Figure 5b) are clustered to Π_i (over-clustering), but less numbers of points (black dots in Figure 5b) are clustered to Π_j (under-clustering) with lower votes as the plane is later selected. In particular, this over- or under-clustering problem produces errors around the boundary as those points, for instance the ones inside dashed circle in Figure 5b, has similar height differences from both Π_i and Π_j . This clustering ambiguity leads to the mis-location of the intersection lines (Figure 5c). To avoid this problem, a post-validation process is conducted before extracting intersection lines. First, points located around the boundary between adjacent planes are collected, and then orthogonal distances from each boundary point to adjacent planes are measured. A true planar membership of the point is now determined by selecting the plane having the shortest distance measure of adjacent planes. This process continues until no boundary points change their planar membership. By re-estimating the result of planar clustering, more accurate intersection line computation can be obtained (Figure 5d).

Binary Space Partitioning

A key issue of modeling 3D building objects is to recover the geometrical and topological relations between adjacent planar or linear segments that are usually fragmented or missed. Instead of pre-specifying a set of explicit relations of adjacent features as *a priori* knowledge, this section employs the well-known data structure of the BSP tree as a mid-level feature grouping tool for object reconstruction.

The BSP is a method for recursively subdividing n -dimensional space into convex sets with homogeneous property by $(n-1)$ -dimensional hyperplane. A consequence

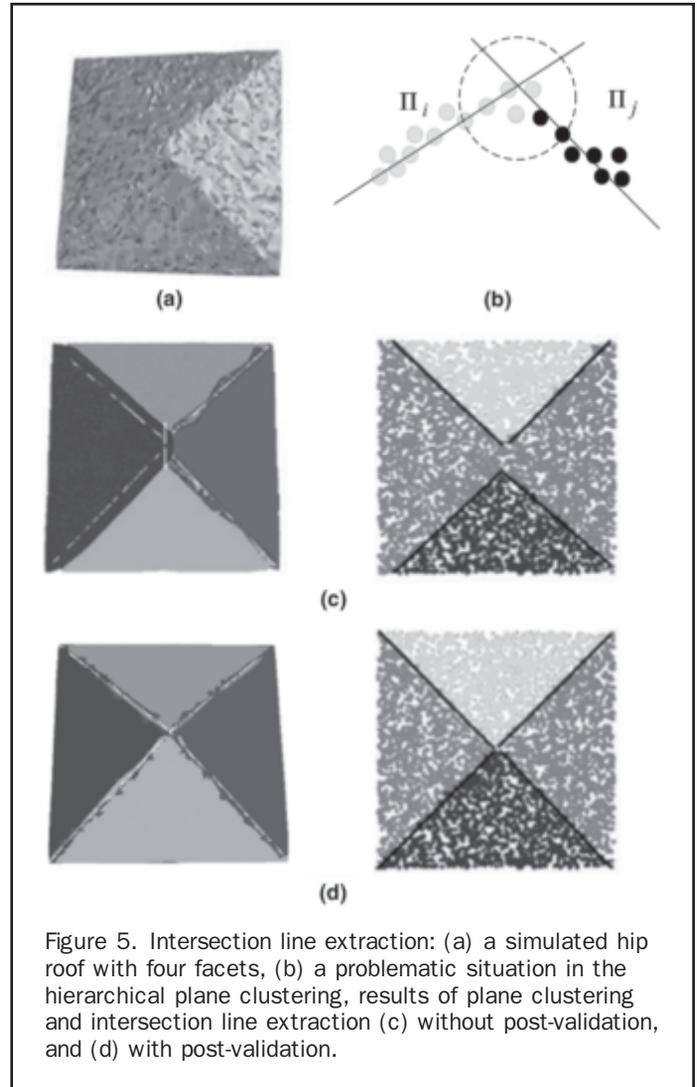


Figure 5. Intersection line extraction: (a) a simulated hip roof with four facets, (b) a problematic situation in the hierarchical plane clustering, results of plane clustering and intersection line extraction (c) without post-validation, and (d) with post-validation.

of this hierarchical subdivision is formulated as a representation of the complex spatial scene by means of a binary tree data structure. The BSP tree is a widely used representation of solid geometry, and is extremely versatile due to its powerful classification structure. The most traditional application of the BSP tree is in 3D computer graphics in order to increase rendering efficiency including solid modeling, hidden surface removal, shadow generation, visibility orderings, and image representation (Fuchs *et al.*, 1980; Gordon and Chen, 1991; Naylor, 1990). Instead of utilizing the tree structure as a pre-processor to store a virtual environment comprising polygonal solid models, Sohn and Dowman (2001 and 2007) has presented potentials of the BSP tree as an object reconstruction tool to produce prismatic models, either using monocular high-resolution satellite, or by combining it with airborne lidar data. The methods were successfully applied to urban city models, but limited to the reconstruction of building outlines with no detailed description of building rooftop structures. In this section, we further exploit the BSP tree for reconstructing complex rooftop models by involving new types of modeling cues including linear features and plane clusters comprising the shape of building rooftop. To deal with increased numbers of modeling cues with different properties, we investigate a new partitioning function and strategy to obtain the

optimality in the generation of the BSP tree. The BSP tree serves as a special data structure consisting of convex hierarchical decompositions of 3D lidar points. With this tree structure, a polyhedral building model is represented by the union of convex regions, which correspond to segmented planar roof-facets that are delineated by a rectilinear boundary with maximum response to real edges. In other words, the convex decompositions of the lidar space induced by the BSP method serve as a fusion framework for integrating area-features (i.e., planar clustering result) and edge-features (line extraction result) for representing the boundary of 3D building rooftops.

Figure 6 illustrates a general idea for the construction of the BSP tree. As discussed earlier, the rectangle bounding each building object is given as the initial building model P_0 . The entire member points of P_0 are labeled, shown by different colors in Figure 6, based on the planar clustering result. Both step and intersection lines $\{l_i\}$ extracted by the line detectors are formulated as the hyperlines $\{h_i\}$, each of which will be involved in the subdivision of the given building polygon, which is described by:

$$h_i(\rho_i, \theta_i) = \{(x, y) \in R \mid x \cos \theta_i + y \sin \theta_i - \rho_i = 0\} \quad (4)$$

where (ρ_i, θ_i) means the distance of the origin from a line segment l_i , and the slope angle measured between the edge normal and x-axis, respectively. A hyperline h_i is then chosen to halve planar clustered points into the positive and negative region, P_{i+} and P_{i-} such that at least one of the two polygons retains relatively higher planar homogeneity and more favorable geometric properties than the ones produced by the other hyperlines; more detailed criteria for the hyperline selection will be described later in this section. Each of sub-polygons is expressed by:

$$\begin{aligned} P_{i+}(h_i; \rho_i, \theta_i) &= \{(x, y) \in R \mid x \cos \theta_i + y \sin \theta_i - \rho_i > 0\} \\ P_{i-}(h_i; \rho_i, \theta_i) &= \{(x, y) \in R \mid x \cos \theta_i + y \sin \theta_i - \rho_i < 0\} \end{aligned} \quad (5)$$

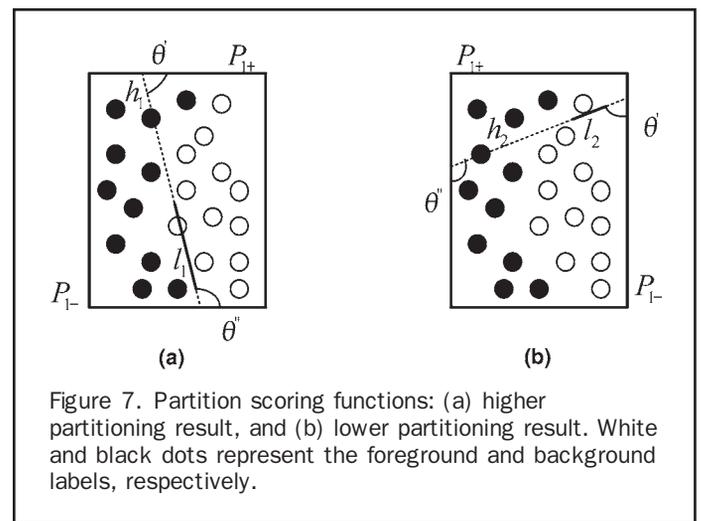
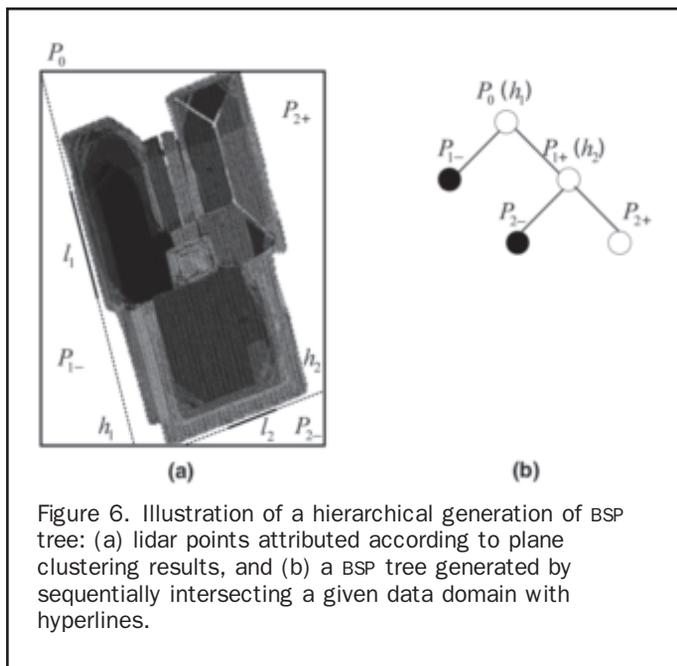
The normal vector of the hyperline h_i is defined by $(\cos \theta_i, \sin \theta_i)$. The positive polygon P_{i+} corresponds to the one that lies in the direction of the normal vector of l_i , the

negative polygon P_{i-} is located in the opposite direction. A BSP tree is now constructed as the root node holds the hyperline h_1 , where all vertices comprising P_0 and the two sub-polygons are represented by leaf nodes. Each leaf of the BSP tree contains binary attributes, closed and open polygon, according to a degree of co-planar homogeneity over a given polygon. A convex region will be represented as the open polygon (node) in the BSP tree if, within the polygon, the member points are attributed with more than one planar cluster and also any part of the line segments is found; in other cases, the node will be attributed as the closed polygon. This node classification determines a triggering and terminating condition of the BSP over the node. That is, when a successive hyperline h_1 is selected as presented in Figure 7, the line continues to partition an un-partitioned (open) polygon P_{i+} . However, the partitioning process will be terminated to a closed polygon P_{i-} . This process continues until no open leaf is generated by BSP.

The partitioning result will be different when a different sequence of line segments is employed. Thus, it is necessary to have the hyperline selection criterion which provides an optimal partitioning result over P_i . This optimality is achieved by the hypothesize-and-test scheme with a partition scoring function. The partitioning score for a hyperline h_i is measured by three criteria: (a) plane homogeneity, (b) geometric regularity, and (c) edge correspondence.

Plane Homogeneity

This criterion controls hierarchical generation of the BSP tree by maximizing the numbers of points with similar planar properties for partitioned polygons. Suppose that an arbitrary polygon P_i contains points with N different plane labels $L_p - \{L_i; i = 1, \dots, N\}$ by the plane clustering method presented in previous section. A representative plane label of P_i is determined as L_r to which maximum numbers of plane labels of P_i are assigned. The plane labels L_p are then binarized into either foreground label L_{fore} or background label L_{back} . A member point of P_i is labeled as L_{fore} if its plane label corresponds to L_r , otherwise as L_{back} . As an arbitrary hyperline h_i partitions P_i into two sub-polygons P_{i+} and P_{i-} , a higher score of the plane homogeneity is given for h_i if maximum labeling homogeneity of both L_{fore} and L_{back} is



obtained in each partitioned sub-polygon. This normalized strength of the plane homogeneity SH is measured by:

$$SH(P_i; h_i) = \max(SH(P_{i+}; h_i) (SH(P_{i-}; h_i)))$$

$$SH(P_{i+}; h_i) = \frac{1}{2} \left\{ \frac{N_{fore}(P_{i+}; h_i)}{N_{fore}(P_i; h_i)} + \frac{N_{back}(P_{i+}; h_i)}{N_{back}(P_i; h_i)} \right\}$$

$$SH(P_{i-}; h_i) = \frac{1}{2} \left\{ \frac{N_{fore}(P_{i-}; h_i)}{N_{fore}(P_i; h_i)} + \frac{N_{back}(P_{i-}; h_i)}{N_{back}(P_i; h_i)} \right\} \quad (6)$$

Edge Correspondence

A better partitioning result can be achieved when the boundary between two partitioned regions strongly corresponds to real edges. The strength of edge correspondence SE is measured by the ratio of lengths of a physical line l_i extracted from lidar data and the corresponding hyperline h_i . A higher score for the edge correspondence is assigned to h_i if a longer length of l_i is found in the polygon by

$$SE(P_i; h_i, l_i) = \frac{Length(l_i)}{Length(h_i)} \quad (7)$$

Geometric Regularity

Most building models have regular geometry (i.e., orthogonal, parallel, symmetric properly), rather than sharp corners. This heuristic preference on the geometric regularity SG is measured by the minimum intersecting angle between P_i and h_i . A lower score is given for h_i when h_i intersects P_i with sharper intersecting angles; scores increase as the minimum intersecting angle increases. Note that the intersecting angles are measured only for one of two child polygons of P_i which contains larger numbers of the foreground labels (Figure 7);

$$SG(P_i; h_i) = Ang(P_i; h_i)$$

$$Ang() = [0(0^\circ \leq \vartheta \leq 30^\circ), 0.5(30^\circ \leq \vartheta \leq 60^\circ), 1(60^\circ \leq \vartheta \leq 180^\circ)] \quad (8)$$

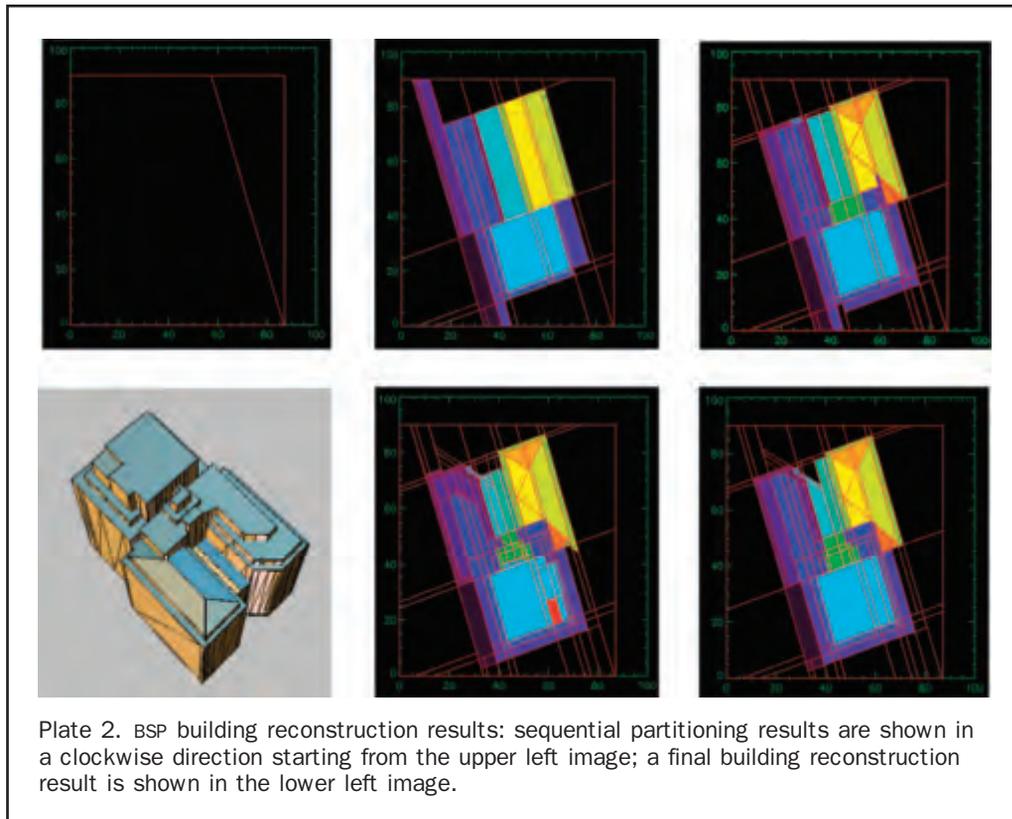
All the hyperlines are tested to partition P_i , and the partitioning result generated by each hyperline is evaluated by a partitioning scoring function. A hyperline, h^* , with the highest partitioning score is finally selected to partition P_i by:

$$h^* = \arg \max_{\forall(h)} (\alpha \times SH(P_i; h_i) + \beta \times SL(P_i; h_i, l_i) + \gamma \times SG(P_i; h_i)) \quad (9)$$

where $\alpha + \beta + \gamma = 1$ and usually (α, β, γ) is chosen as (0.5, 0.3, 0.2), respectively. After constructing the BSP tree, a plane adjacency graph is created by collecting final leaves of the BSP tree where each node represents a planar roof-facet and each arc represents the connectivity between neighboring planes. Starting from the node with the largest area in the plane adjacency graph, a simple validation of normal vector compatibility is applied to its adjacent planes. The planes having similar normal vector angles are merged and planar parameters for merged planes are re-estimated. This merging process continues until no plane can be accepted by the co-planar similarity test. Once all polygons are merged together, building boundaries are reconstructed. Plate 2 shows a sequence of the building reconstruction results obtained by the presented method based on the BSP tree.

Experimental Results

In this section, we discuss and evaluate the performance of the building reconstruction technique we propose here. Figure 8a shows an elevation model over downtown Toronto. The data was acquired with an Optech Incorporated ALTM-2050 airborne laser scanner with a pulse



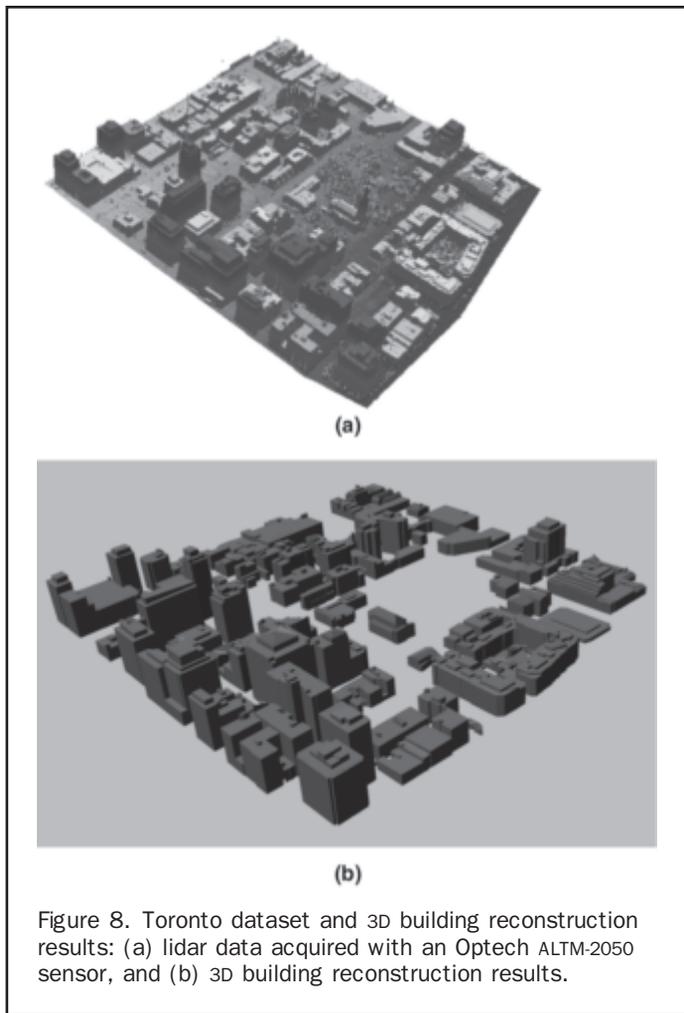


Figure 8. Toronto dataset and 3D building reconstruction results: (a) lidar data acquired with an Optech ALTM-2050 sensor, and (b) 3D building reconstruction results.

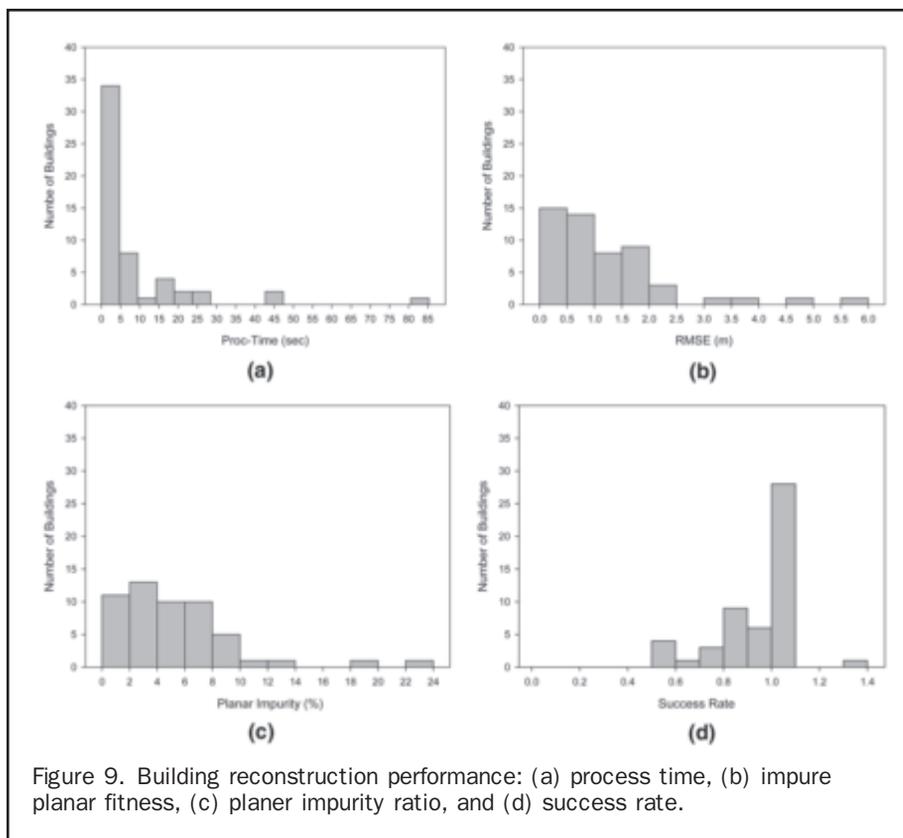
repetition rate of 50 KHZ at a flying height of 850 m above ground level. The Toronto test dataset covers approximately 1 km by 1 km where a total 2,741,684 laser points were acquired with a mean density of 2.74 points/m² (~0.6 m point spacing). According to the sensor and flight specifications, 0.43 m horizontal and 0.15 m vertical accuracies are expected. The overall elevation is almost flat, and the northern part of the data is slightly higher (approximately 100m) than the southern area. The test scene was chosen to include many buildings with different orientations and roof structures; consequently it is a good region to test our reconstruction algorithm.

Figure 8b shows a final result of 3D building reconstruction from the Toronto dataset based on the proposed method. As a prerequisite processing step for achieving the result of Figure 8b, all the building objects were isolated using the building detection procedure described in the previous section. As a consequence, a total of 53 buildings comprising 182,450 points were detected, each of which was bounded by an initial building model (i.e., rectangle). All the member points in each rectangle were attributed as non-building and building points, and were sequentially clustered by height similarity and planar similarity, through which step and intersection lines were extracted. As those initial models were hierarchically partitioned by extracted rectilinear lines, convex polygons representing roof-facets were produced with finer scales and rooftop models were reconstructed as similar roof-facets were merged. All of the parameters used for this experiment were described in the previous section.

In Figure 8b, a total of 529 roof-facets were reconstructed. Since the Toronto dataset includes buildings with a wide range of shape complexity, the number of roof facets reconstructed varies from one to 43 and an average of 10 planes per building with the standard deviation of 12.3 planes. Out of a total 182,450 points, 94.48 percent (172,378 points) were recognized as *planar* points by the planar clustering algorithm (i.e., representing a particular planar roof-facet), while 5.52 percent (10,072 points) were not clustered as the *planar* points, but were eliminated during the planar clustering process. Also, 87.7 percent (160,078 points) of the *planar* points were actually utilized for the final reconstruction of the roof-facets as the foreground points (those points are termed *pure planar* points), while 5.5 percent *planar* points (10,072 points), namely *impure planar* points, remained as the background points in under-partitioned roof-facets. We evaluated the overall *pure* accuracy of the roof-facet reconstruction by measuring the residual of the *pure* planar points from a plane reconstructed in the Root Mean Squared Error (RMSE). The *pure* plane reconstruction accuracy was measured as the mean $\mu = 0.38$ (m), and standard deviation $\sigma = 0.22$ (m).

The proposed algorithms were implemented and performed on a Desktop PC with a Pentium-4 processor at a clocked speed of 2.40 GHZ and with 512 MB RAM. We attempt to evaluate the efficiency of our method by measuring the total time consumed in reconstructing the entire test scene. The execution time of building reconstruction on this platform was a total of 461.1 seconds, which corresponds to 8.5 seconds per building. The most of buildings (79 percent buildings) were reconstructed in less than 10 seconds, while the maximum execution time reached to 85.3 seconds over a building, with the largest area of 8,270 m² and the maximum number of planes (43 planes). Note that this figure does not include all the time taken for building detection. (Figure 9a)

Since the ground truth of 3D building rooftops was not available over the test site, an absolute comparison of reconstruction accuracy between the ground truth and reconstructed polygons was not possible. Thus, we examined the accuracy of the building reconstruction results in two relative measures, by measuring the residuals of the *impure planar* points from a reconstructed plane (*impure planar fitness*) and the ratio of the number of the *impure planar* points against the total numbers of *pure planar* points in a roof-facet reconstructed (*planar impurity*). This plane reconstruction accuracy cannot estimate in an absolute scale the geometric accuracy and the topological consistency of the reconstructed roof polygons. However, these measures can indirectly indicate an overall quality of the reconstruction performance, in a sense that lower residuals, and a lower ratio of *impure planar* points can be achieved as the roof polygons are reconstructed by maximizing the planar homogeneity, and thus with the minimum numbers of the *impure planar* points. If the reconstructed building model does not preserve original building boundaries or misses some part of building structures, more numbers of the *impure planar* points are produced and thus, both measurements are increased. Figure 9b and 9c shows histograms of the measurements of the *impure planar fitness* and the *planar impurity* respectively. The RMSE of the *impure planar fitness* for all the buildings was measured as 1.1 m, and $\sigma = 1.2$ m, while the average ratio of the *planar impurity* was measured as 5.1 percent with $\sigma = 4.4$ percent. As described earlier, the amount of the *impure planar points* increases both measures. It was observed that most of the *impure planar points* were produced around roof boundary regions where adjacent roof-facets were not correctly delineated by reconstructed models due to geometrically erroneous hyperlines or over roof-facets which were under-partitioned due to a lack



of hyperlines extracted. A large *impure planar fitness*, higher than 3 m, were found over four buildings as planar roof-facets comprising those buildings are connected each other with large height differences.

Finally, we also evaluated the reliability of our building reconstruction system by comparing the number of automatically reconstructed roof-facets to the one that was manually counted. The success rate of the algorithm is defined by:

$$R = 1 - \frac{N_t - N_r}{N_t}. \quad (10)$$

Here, N_r is the total number of planes reconstructed by our method, and N_t is the total number of planes that are visually found by a human operator. In the tests, accuracy is only judged by the human operator with a comparison to the manual results. The success rate R will be larger than 1 if the algorithm over-reconstructs building roof-facets, but is less than 1 if roof-facets are under-reconstructed by the method. A perfect success rate of 1 will be achieved if no roof plane is missed by our method, but R will be 0 if the algorithm cannot reconstruct any roof-facet at all. Figure 9d shows the statistical result of the test of the reliability of the automatic building reconstruction method. The success rate R is measured as $\mu = 0.92$ ($\mu_{\min} = 0.5$ and $\mu_{\max} = 1.3$) with $\sigma = 0.2$. The experiments indicate a great potential for the developed method for practical GIS data production. This suggests that when a building is comprised of 10 roof-facets, our method fails to reconstruct one plane. As seen in the figure, the algorithm tends to under-reconstruct building rooftops, rather than over-reconstruct the models. It has been visually confirmed that the over-reconstruction of building models occurred in cases when either a small flat roof or super-structure was hard to distinguish in

height from the neighboring large-area planes during the height clustering process; or step lines, usually having a short length of 1 m to 3 m, could not be detected by the edge detector due to high irregular density of lidar data.

In Figures 10, 11, and 12, we visualized several examples of 3D buildings that had been reconstructed in order to discuss the characteristics of the proposed building reconstruction system. Table 1 shows the information of presented buildings and reconstructed models. The most of prismatic buildings with single flat roofs were reconstructed with the highest performance. For instance, two buildings in Figure 10 show a simple “T” and a rectangular shape. A slightly higher performance in the planar fitness and the planar impurity was measured for the building in the second row of Figure 10b in which there are more numbers of roof super-structures. It was confirmed that most of the prismatic buildings with one or two planar roof facets show a similar level of planar fitness and impurity. This is because low-level modeling cues of planar clustering and linear features can be extracted with higher quality (i.e., less fragmentary or missing levels with higher geometric accuracy). It is worthwhile to point out that the developed reconstruction method can deal with irregular deficiency of laser-scanning acquisition. It was observed that there are many regions occluded by building objects so that lidar data are missing near to the building roof boundary. As seen in the second row of Figure 10b, the left-hand side of the building in the lidar data was a little intruded from the original rectangular shape since the lidar data were missing due to the occlusion effect. Nevertheless, there were no missing parts in 3D building model reconstructed. This is one of the advantages as the reconstruction algorithm is directly applied to irregular points, rather than to the gridded format.

In Figures 11 and 12, the buildings show more complicated shapes than in Figure 10, where several building primitives are horizontally or vertically connected to each other. It is

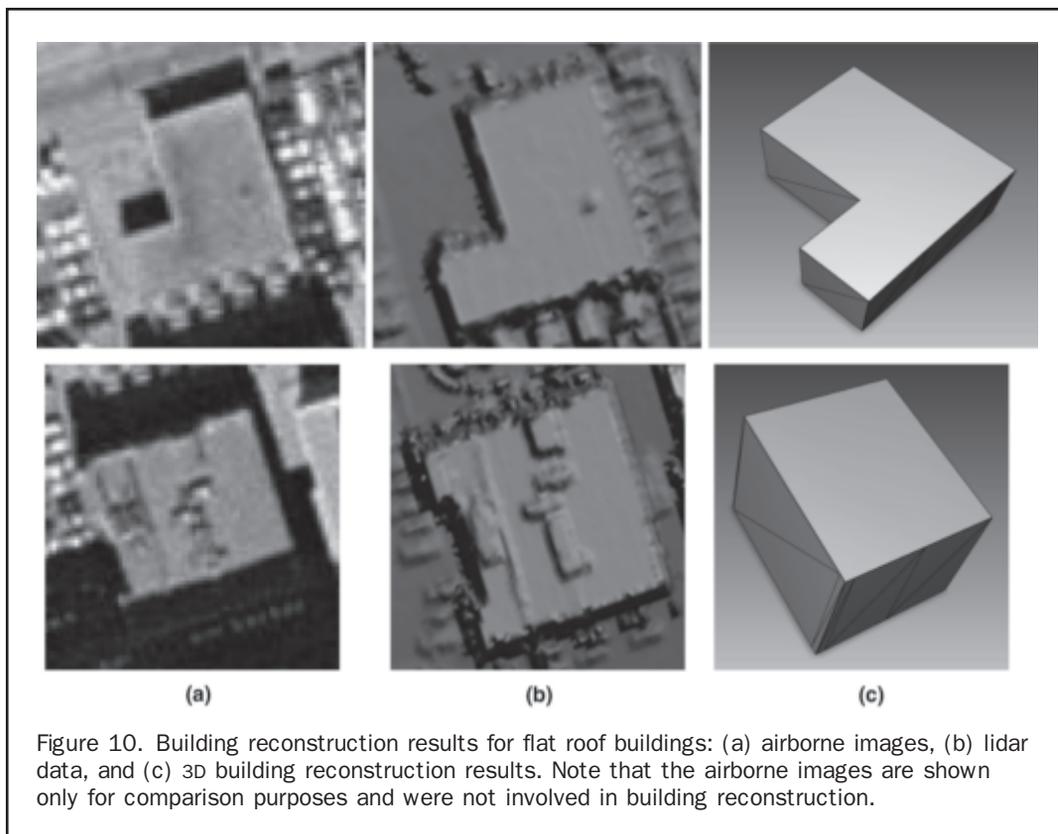


Figure 10. Building reconstruction results for flat roof buildings: (a) airborne images, (b) lidar data, and (c) 3D building reconstruction results. Note that the airborne images are shown only for comparison purposes and were not involved in building reconstruction.

evident that all of the buildings in the figures contain large amounts of data acquisition errors due to the small coverage. These errors may make it difficult to recognize some of the rooftop structures and sharp boundaries. Although the shape complexity and data irregularity have increased, Figures 11 and 12 showed that the overall quality of the reconstruction result based on the proposed technique is satisfactory, and that the reconstructed models effectively preserve the original geometry. In fact, 3D structures of the buildings presented in these figures were not very easily interpreted by visual investigation, either from the optical imagery or lidar data. Also the reconstruction results were achieved without requiring *a priori* knowledge of specific building models or of the topological relations between adjacent planes. Thus, it can be concluded that the developed techniques would be useful to quickly provide an initial, yet fine rooftop model with very high shape complexity. However, it is also evident that the reconstructed models are not perfect. As seen in the first row of Figure 11, details of the structures around the corners were lost and most of them were generalized too much as rectangular corners. Also, a super-structure in the middle of the building was wrongly reconstructed, although the original shape of this structure is not certain, even by visual investigation. Those problems may be caused by a significant lack of point density covering those particular regions, which results in missing linear features.

Summary and Conclusions

We have proposed and presented a new algorithm for automatic 3D reconstruction of building rooftops from airborne lidar data. In the current research, this task has been achieved through the development of a unique data-driven framework that employs the BSP tree algorithm. The BSP-based method was used to reconstruct polyhedral building models by implicitly grouping fragmentary linear features

between adjacent roof-facets. The results show that the method can represent highly complicated building structures, where multiple building primitives are horizontally and/or vertically connected with, and occluded from, each other. Starting from a coarse description, the developed techniques incrementally refine the model at different modeling scale by maximizing the co-planarity, edge correspondence and geometric regularity. By this nature of global optimization, the system can simultaneously reconstruct a large number of connected roof-facets, but does not require a local analysis of the primitive connectivity between adjacent facets that are independently generated. These characteristics are the most beneficial compared to existing model-driven techniques which may suffer difficulties under these circumstances. Another advantage of BSP algorithm, which has not been studied yet in current analysis, is the ability to combine a search structure with a representation scheme into one unique data structure. For instance, a building model can be represented in a different scale, i.e., by controlling the depth of BSP tree. This useful property can be further exploited for generalizing building shapes according to the level of detail (LoD). Due to the irregular nature of laser scanning and building occlusion problems, it is usually difficult to provide a complete set of linear features for representing building models from lidar data. The proposed BSP algorithm provides an alternative approach to the traditional feature grouping approaches to overcoming the difficult problem of when linear features are fragmented into an unpredictable order.

Although the method was successfully applied for reconstructing buildings from lidar, there are still some drawbacks that indicate our future work could be extended in order to improve the method:

- The modeling accuracy of roof-facets reconstructed based on our method is mainly subject to the extraction quality of linear features. Improper location of extracted linear features leads to unsatisfactory or erroneous results. Also, some parts

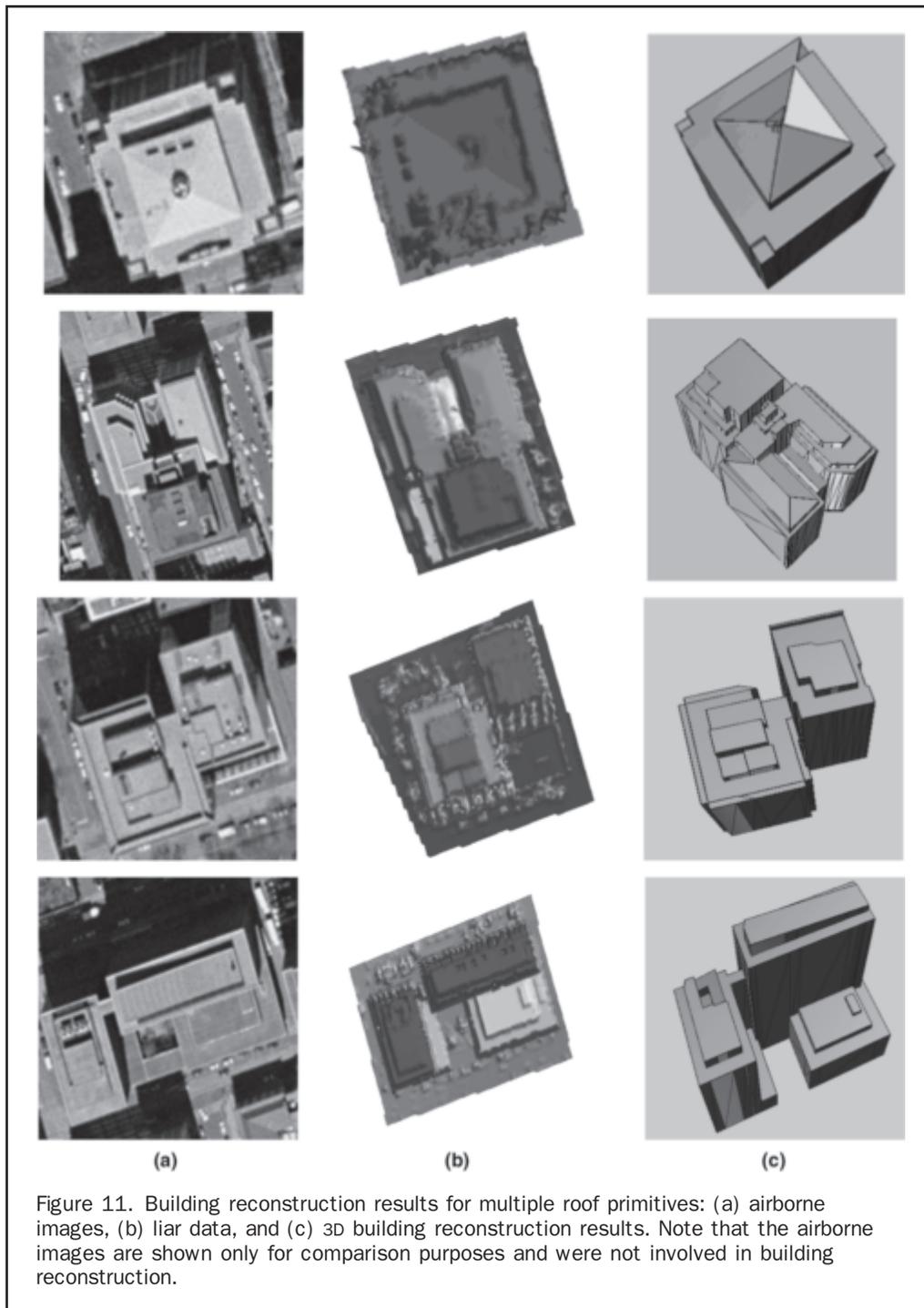


Figure 11. Building reconstruction results for multiple roof primitives: (a) airborne images, (b) LiDAR data, and (c) 3D building reconstruction results. Note that the airborne images are shown only for comparison purposes and were not involved in building reconstruction.

of building shapes are under-modeled if linear features over those regions are not detectable. During the current experiments, these problems have been observed over the region where the irregularity of point density is so high that some parts of buildings are not well represented by the laser acquisition. One solution to the problem is to devise a compensatory process for recovering missing features of interest by partly combining additional geometric information driven from specific building models. For instance, a hypothetical prediction of symmetrical shape can locate a new conjugate line feature as suggested by Sohn and Dowman (2007).

- In general, height clustering is effective for decomposing a group of connected buildings into simpler regions where the

complexity of planar clustering problems can be reduced. However, as discussed in the previous section, we observed that some small flat roof-facets were not isolated from adjacent ones using this process. This problem is due to the height difference of those roof-facets from the entire building is too weak to be detected by a pre-fixed bin size of the height histogram. As a consequence, 3D roof modeling over those regions fails, which decreases the success rate. To deal with these problems, we need to develop a more adaptive algorithm for the height clustering to varying heights. This will help in reconstructing more building structures with small scales.

- In the current experiment, a geometric regularization has been devised by quantizing line slopes in a limited number

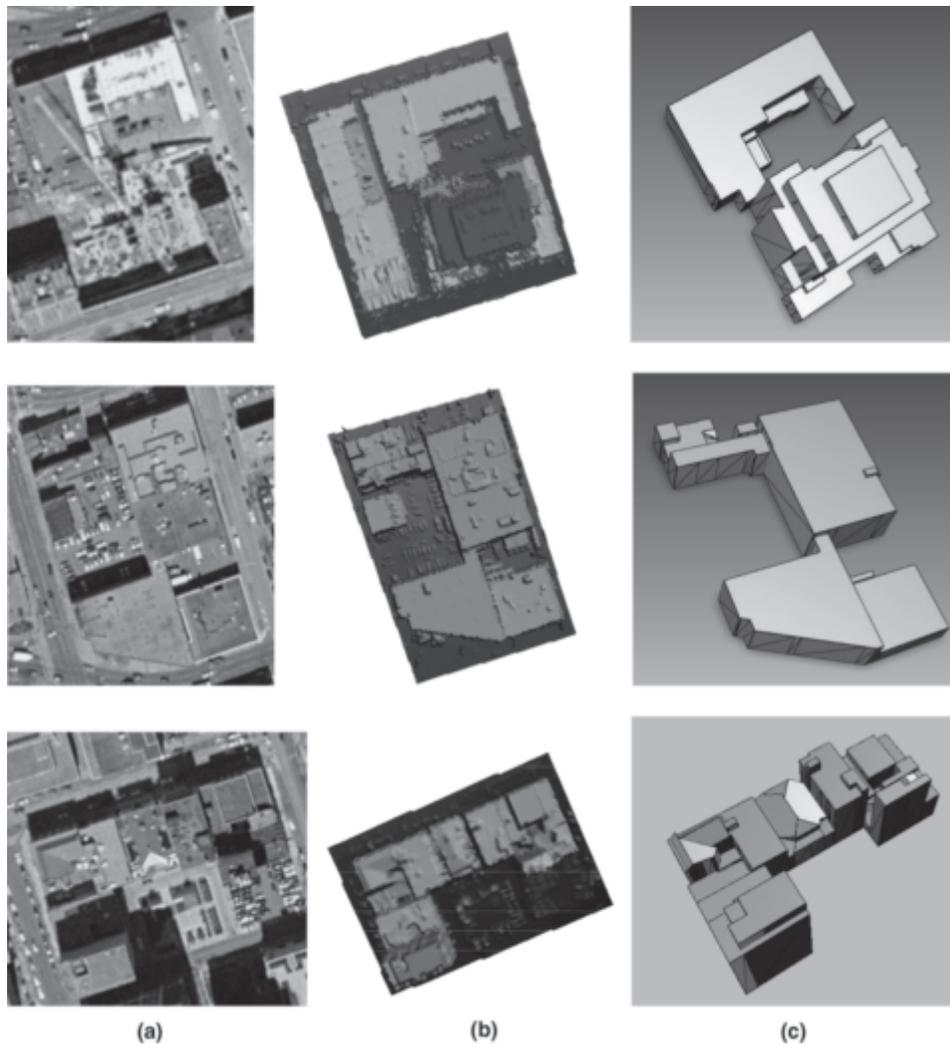


Figure 12. Building reconstruction results for buildings with large area: (a) airborne images, (b) lidar data, and (c) 3D building reconstruction results. Note that the airborne images are shown only for comparison purposes and were not involved in building reconstruction.

of angular ranges during the early stage of feature extraction. However, incorrect reconstruction of building models with irregular and sharp corners was still present in current experiments in case that a few erroneous lines were inevitably involved in the BSP partitioning process. Thus, a future research direction towards this problem, a post-processing procedure to refine data-driven models produced by BSP by combining parametric models will be conducted.

References

- Alharthy, A., and J. Bethel, 2004. Detailed building reconstruction from airborne laser data using a moving surface method, *International Archives of Photogrammetry and Remote Sensing*, 35(Part B3):85–92.
- Ameri, B., 2000. *Automatic Recognition and 3D Reconstruction of Buildings from Digital Imagery*, Ph.D. dissertation, Institute of Photogrammetry, University of Stuttgart, 110 p.
- Baillard, C., and H. Maitre, 1999. 3D reconstruction of urban scenes from aerial stereo imagery: A focusing strategy, *Computer Vision and Image Understanding*, 76(3):244–258.
- Brenner, C., 2000. Towards fully automatic generation of city models, *International Archives of Photogrammetry and Remote Sensing*, 33(Part B3):85–92.
- Brenner, C., and N. Haala, 1998. Fast production of virtual reality city models, *International Archives of Photogrammetry and Remote Sensing*, 32(4):77–84.
- Brenner, C., 2005. Building reconstruction from images and laser scanning, *International Journal of Applied Earth Observation and Geoinformation*, 6(3–4):187–198.
- Chen, S., and D. Gordon, 1991. Front-to-back display of BSP trees, *IEEE Computer Graphics and Applications*, 11(5):79–85.
- Filin, S., 2002. Surface clustering from airborne laser scanning data, *International Archives of Photogrammetry and Remote Sensing*, 34(Part 3):119–124.
- Fuchs, H., Z.M. Kedem, and B.F. Naylor., 1980. On visible surface generation by a priori tree structures, *Computer Graphics*, 14(3):124–133.
- Hofmann, A.D., 2004. Analysis of TIN-structure parameter spaces in airborne laser scanner data for 3-D building model generation, *International Archives of Photogrammetry and Remote Sensing*, 35(B3):302–307.

TABLE 1. BUILDING RECONSTRUCTION PERFORMANCE

	Area (m ²)	Nr	Nt	Impure Planar Fitness	Planar Impurity	Success Rate (R)
Figure 10a	279.63	1	1	0.0967	0	1
Figure 10b	526.73	1	2	0.6684	0.1030	0.5
Figure 11a	2,116.59	10	10	0.9504	4.5141	1
Figure 11b	3,269.09	23	25	1.9776	13.6801	0.92
Figure 11c	3872.45	9	9	4.6757	9.2088	1
Figure 11d	2,329.86	16	14	0.8770	2.5537	0.88
Figure 12a	3,486.09	15	18	3.1449	3.3127	0.83
Figure 12b	5,794.29	15	11	0.6208	2.3137	0.80
Figure 12c	4,329.66	38	42	2.2703	7.3140	0.91

- Hu, Y., 2003. *Automated Extraction of Digital Terrain Models, Roads and Buildings Using Airborne Lidar Data*, Ph.D. dissertation, University of Calgary, Calgary, Alberta, 206 p.
- Maas, H.-G., and G. Vosselman, 1999. Two algorithms for extracting building models from raw laser altimetry data, *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3):153-163.
- Rottensteiner, F., J. Trinder, S. Clode, and K. Kubik, 2005. Using the Dempster-Shafer method for the fusion of LIDAR data and multi-spectral images for building detection, *Information Fusion*, 6(4):283-300.
- Schwalbe, E., H.-G. Maas, and F., Seidel, 2005. 3D building model generation from airborne laser scanner data using 2D GIS data and orthogonal point cloud projections, *International Archives of Photogrammetry and Remote Sensing*, 36(Part 3/W19):209-214.
- Sohn, G., and I.J. Dowman, 2001. Extraction of buildings from high resolution satellite data, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)* (A. Grün, E.P. Baltsavias, and O. Henricsson, editors), Balkema Publishers, Lisse, pp. 345-356.
- 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.
- Sohn, G., and I. Dowman, 2008. A model-based approach for reconstructing terrain surface from airborne LiDAR data, *The Photogrammetric Record*, 23(122):170-193.
- Suveg, I., and G. Vosselman, G., 2004. Automatic 3D reconstruction of buildings from aerial images, *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3-4):202-224.
- Vosselman, G., 1999. Building reconstruction using planar faces in very high density height data, *International Archives of Photogrammetry and Remote Sensing*, 32(Part 2):87-92.
- Vosselman, G., and S. Dijkman, 2001. 3D building model reconstruction from point clouds and ground plans, *International Archives of Photogrammetry and Remote Sensing*, 33(3/W4):37-44.
- 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*, 36(8/W2):33-38.
- Weidner, U., and W. Förstner, 1995. Towards automatic building reconstruction from high resolution digital elevation models, *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):38-49.

(Received 29 January 2007; accepted 11 May 2007; revised 06 September 2007)