AUTOMATIC EXTRACTION OF LARGE COMPLEX BUILDINGS USING LIDAR DATA AND DIGITAL MAPS

Jihye Park [a], Impyeong Lee [a, *], Yunsoo Choi [a], Young Jin Lee [b]

[a] Dept. of Geoinformatics, The University of Seoul, 90 Jeonnong-dong, Dongdaemun-gu, Seoul, Korea - (jihye, iplee, choiys,)@uos.ac.kr

[b] Telematics ·USN Research Division, Electronics and Telecommunications Research Institute, 161 Gajeong-dong, Yuseong-gu, Daejeon, Korea - yjinlee@etri.re.kr

Commission Ⅲ, WG Ⅲ/3

KEY WORDS: LIDAR data, Digital Map, Building Model, Primitives, Patches

ABSTRACT:

As the use of building models are rapidly increased for various applications, many studies have been performed to develop a practical and nearly automatic method to extract such models from various sensory and GIS data. Nevertheless, it is still a difficult problem to extract the models of large-complex buildings in particular. The purpose of this study is thus to develop a fully automatic method to extract the detail models of buildings from LIDAR data and a digital map. This extraction consisting of primitive extraction and modeling is mainly based on robust segmentation of planar patches from numerous LIDAR points. These primary primitives are used as the references to generate secondary primitives such as edges and corners and then refined based on these secondary primitives to form a complete polyhedral model. The proposed method was successfully applied to extracting large-complex buildings from real data in the test site. It can be a promising time- and cost-effective solution for a country to enhance their traditional map to include 3D models of buildings.

1. INTRODUCTION

The need of detail and realistic building models is rapidly increasing because of their intensive uses for various applications not limited to urban planning and redevelopment, three-dimensional car navigation systems, video games, and others areas.

To reconstruct 3D buildings, many studies based on various sensory data have been performed. For examples, using aerial images, Baillard and Zisserman [1] reconstructed polyhedral models using the edges between planar roof patches. The main idea is to obtain the half planes to the left and right of a detected dihedral line segment. The advantage is that only relatively local information is exploited (Brenner, 2003). In recent research, Suveg and Vosselman (2004) reconstructed buildings using aerial images and 2D ground plans. They generated the 3D volumetric primitives using the 3D corners extracted from 2D digital map and filtered by images. 75% of all objects were extracted using this method.

Building reconstruction from LIDAR data are very active these days. Rottensteiner and Briese (2003) extracted roof faces from DSM and derived the intersection and step edges from the regularized DSM. Additionally, images were used to detect small buildings. In recent studies for extracting the roof faces, Lodha and Kumar (2005) applied K-Mean algorithm to refine LIDAR points and to detect the planar roof faces. Since users should assign the number K indicating the number of point clusters, this approach is a semi-automatic method. Vosselman (1999) extracted roof faces from LIDAR points using a Hough transform. Vosselman and Dijkman (2001) improved this method by using ground plans in addition to LIDAR data. Brenner (1998) generated building models from LIDAR data and 2D ground plan using a heuristic algorithm.

Although many researchers have proposed semi-automatic (or automatic) methods, it is not yet solved to extract the detail models of large-complex buildings in particular in a fully automatic manner. In most cases, the modeling processes still have involved intensive manual editing steps and thus been thought to be time and money consuming.

The purpose of this study is thus to develop a fully automatic method to extract the detail models of buildings in particular with large-complex roof structure from LIDAR data and a digital map. This extraction includes two stages, *primitive extraction* and *modeling*. The most important step of this process is segmentation of planar patches. These primary primitives are used as the references to derive the secondary primitives such as edges and corners and refined to form each facet of the complex roof structure.

2. OVERVIEW AND PREPROCESSING

2.1 Overview of the Proposed Method

As the framework shown in Figure 1, the proposed building modeling approach include three main stages, that is, preprocessing input data, extracting building primitives, and generating building models. The inputs are airborne LIDAR data and the building layers of a digital map covering the same area. During the preprocessing stage, the LIDAR data are

[*] Corresponding author

registered to the digital map and the LIDAR points around and inside each building boundary are extracted. From these points are extracted the building primitives such as surface patches, edges, and corners. These primitives are then refined and grouped into a complete polyhedral building model.

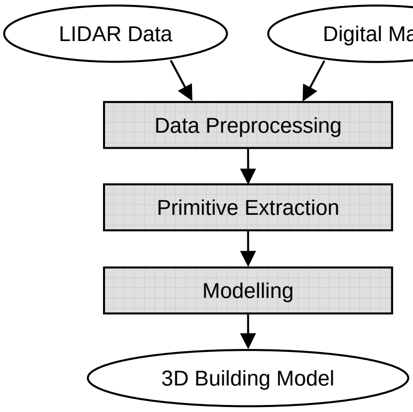

Figure 1. Framework for the building modelling approach

2.2 Input Data

LIDAR data consist of numerous three-dimensional points sampled from the terrain. Although these data provide very accurate elevations of the surfaces, they hardly retain the exact locations of corners and edges of objects in general because of relatively lower sampling density than images. This is a main reason why we also use the building boundary of a large scale map.

The test site is hilly district in Daejun Metropolitan city, Korea. As shown in an aerial image of Figure 2, the site includes many large buildings of various shapes and complex roof structures. The input LIDAR data of this site is shown in Figure 3. The point density is about 5.4 points/m^2. The input digital map is published by National Geographic Information Institute, Korea and the scale is 1/5,000. Figure 4 presents the building layer of this map. The aerial image in Figure 2 has not been used for the input but only for the verification of the modeling results.

Figure 2. Aerial image of the test site

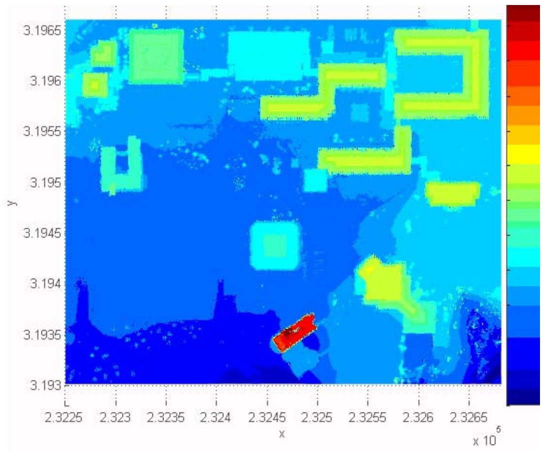

Figure 3. LIDAR data

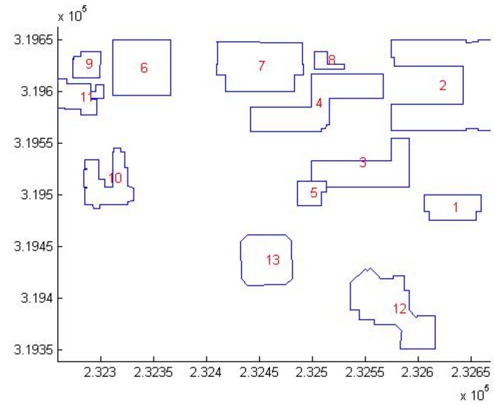

Figure 4. Digital map

2.3 Data Preprocessing

2.3.1 Calibration and outlier elimination: The LIDAR data were calibrated and verified to have better than the accuracy of ± 50 cm and ± 20 cm in horizontal and vertical positions, respectively. Some outliers in the data were detected and removed using the method based on the point density proposed by Moon et. al. (2005). The outlier ratio was found to be about 1 %.

2.3.2 Registration: The LIDAR data were then geometrically registered with the digital map. This registration was performed using the tie points manually selected from both data sets. These tie points mostly locate at the corners of buildings. Since the digital map provides only horizontal coordinates of the corners, only horizontal registration was possible using the 2D similarity transformation. Both data were registered with the precision of about ± 50 cm.

2.3.3 Points Extraction: Two sets of LIDAR points are extracted for each building boundary. One includes the points inside the boundary. We use these inside points to model the roof structure. The other includes the points locating outside the boundary with a distance of less than 5 m from it. The ground along the building can be derived from these outside points. Figure 5 shows the points extracted for all the buildings, where the blue and magenta dots indicates the inside and outside points, respectively.

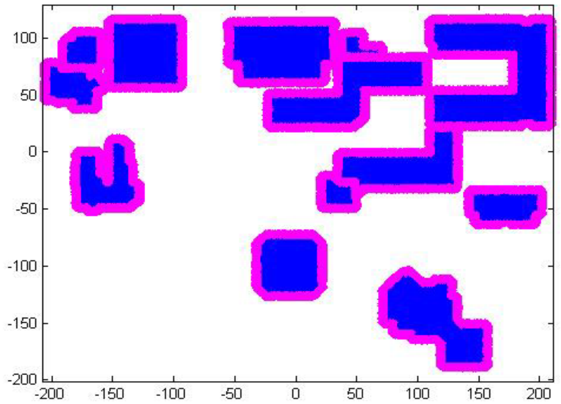

Figure 5. Result of points extraction

3. PRIMITIVE EXTRACTION

The primitives such as planar patches, edges and corners are extracted from the inside and outside points for each building. Planar patches are the *primary primitives* and the others are the *secondary primitives* that can be derived using the primary ones.

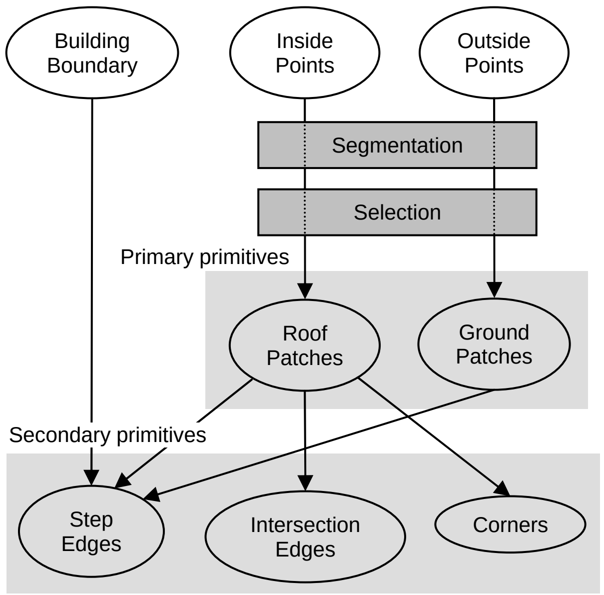

Figure 6. Primitive extraction process

3.1 Patch Adjacency and Connectivity Graph (PACG)

The adjacency and connectivity between all the primitives are examined and the results are stored into a graph structure called *patch adjacency and connectivity graph* (PACG). This graph incorporates each primitive into a node and any identified adjacency between two primitives is stored into an arc between them. Three types of adjacency are defined. The first one is *2D adjacency* created between two primitives if the horizontal distance between their boundaries is less than a given threshold, which is set to 2 m in this study. If the 2D adjacent primitives are also sufficiently close in 3D, *3D adjacency* is established between them. Otherwise, *2D only adjacency* is established. If any two adjacent primitives are verified to be actually connected, *connectivity* is additionally assigned. For example, if a patch is the nearest primitive adjacent to another patch and the intersection edge between them is adjacent to their boundary, they are confirmed to be connected.

3.2 Segmentation of Planar Patches

Planar patches are segmented from the inside and outside LIDAR points, respectively. This segmentation can be performed using the algorithm based on perceptual organization of numerous 3D points. This algorithm was originally proposed by Lee (2002) and summarized as follows:

The segmentation process starts with establishing the adjacency among the LIDAR point irregularly distributed in 3D space. A point cluster is constructed for every point by gathering a small number of points adjacent to the point. Each cluster is approximated to a plane. The clusters with relatively small fitting errors are selected as seed clusters, from each of which a planar patch is then growing with the adjacent points added to the cluster.

During the growing step, every adjacent point to the growing patch is tested about whether the point is statistically consistent with the patch. This growing process for a patch continues until no more adjacent point can pass this test. This grown point cluster called a patch is then verified by checking the size of the cluster and the fitting errors. For the verified patch, its boundaries are computed by determining the outlines of the point cluster using the alpha-shape algorithm (Edelsbrunner, 1983).

After segmentation, the set of points is converted into a set of patches, where each patch is expressed with the plane parameters, the boundary, and the fitting error considered as the roughness.

3.3 Selection of Planar Patches

Some of the patches from the inside points may be unreasonable to be parts of the roof structures. We thus selected as the roof patches only the patches satisfied with the following conditions:

1. The patch size is enough large.
2. Roughness of the patch is relatively low.
3. The shape of the patch is geometrically suitable.

In a similar way, we also select the ground patch among the patches segmented from the outside patches. The selection criteria are as follows:

1. The size of the patch is larger than any other patches.
2. The height of the patch is lower than any other patches.
3. The roughness of the patch is smaller than any other patches.

The adjacency between all the selected roof and ground patches are then examined by checking the distance between any pair of patches. All the patches with the identified adjacency are stored into the PACG.

For example, Figure 7 shows the roof and ground patches selected from the segmented patches for a building (ID: 2). As compared with its representation given in the aerial image and the LIDAR plots in Figure 2 and 3, the segmented and selected patches reasonably describe the roof structure and the ground around the building.

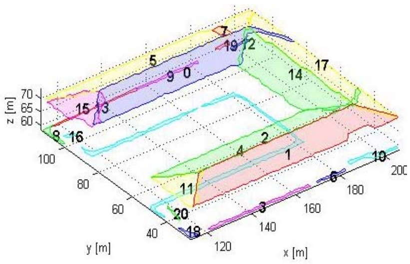

Figure 7. Roof and ground patches

3.4 Intersection Edges

Intersection edge can be derived from two 3D adjacent patches using the algorithm as follows:

1. Select a pair of adjacent patches identified from PACG.
2. Compute a straight line intersected by the planes.
3. Compute the distance between this line and the boundary of each patch.
4. If the distances become greater than a given threshold, discard the line.
5. Otherwise, determine the two ending points limiting the straight line.

To determine the ending points, we first identify the parts of the boundaries that are adjacent to the straight line and then project them to the straight line. The extreme two points of the range on the straight line in which the parts of boundaries are actually projected are selected as the ending points. The straight segment limited by these points is called intersection edge. This edge is also stored into the PACG with the connectivity assigned to the two adjacent patches. Examples of these edges are shown in Figure 8.

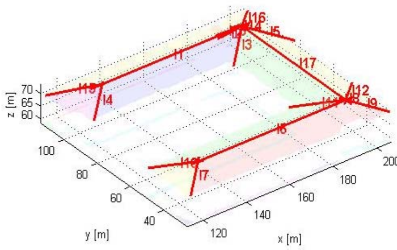

Figure 8. Intersection edges

3.5 Corners

If three patches are identified to be adjacent each other based on the PACG, a corner can be derived from them. With the three planes, an intersection point is computed. If this point is also adjacent to the three patches, it is confirmed as a corner. This corner is also stored into the PACG with the connectivity assigned to the three adjacent patches. Figure 9 shows the derived corners.

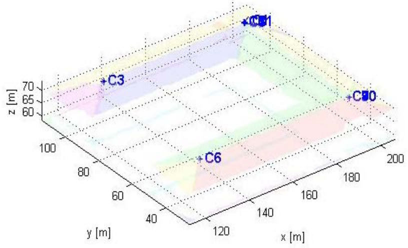

Figure 9. Corners

3.6 Step Edges

Step edges mean the parts of building outlines showing abrupt change in elevation across the edges, for examples, the outlines of vertical walls. These edges are mainly observed along the building boundary provided by the digital map but sometimes inside the roof structure within the boundary. The edges along the boundary are derived by projecting the 2D building boundary to the nearest adjacent roof patches and ground patches. In addition, we derive the edges within the roof structure from any pair of 2D only adjacent patches identified from PACG. The derived step edges are also stored into the PACG with proper connectivity and adjacency assigned. Figure 10 shows examples of the derived step edges.

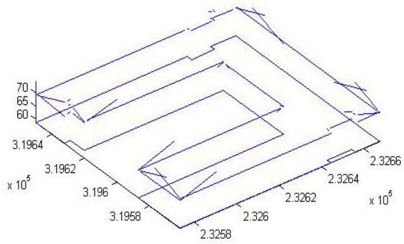

Figure 10. Step edges

4. MODELING

The derived roof patches as the primary primitives has also important roles in the modeling process shown in Figure 11. Their boundaries are refined using the secondary primitives such as the corners, the intersection edges, and the step edges. The refined roof patches are then called *roof facets*. The space between each pair of the step edges connected in 2D can be filled with a vertical planar patch, dedicated to *wall facets*. The roof and wall facets constitute the final polyhedral model.

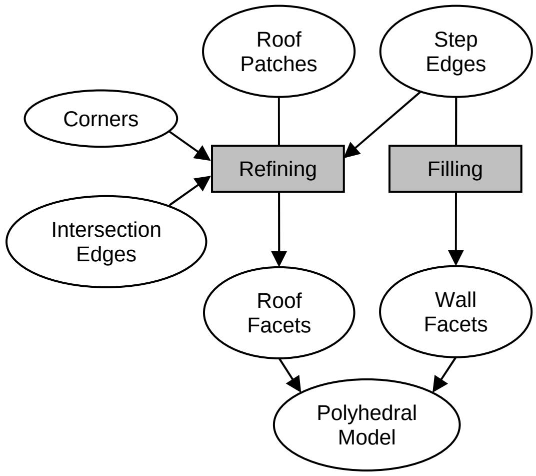

Figure 11. Modelling process

4.1 Roof Facets

Since the boundary of each roof patch is formed by the outer points of the patch, it is so rough that the patch could not be used as a roof facet directly. It is thus necessary to refine the boundaries of roof patches using the secondary primitives located with better accuracy in general.

This refining algorithm is illustrated with a simple example in Figure 12 and summarized as follows:

1. Select a roof patch.
2. Select an edge forming the boundary called a boundary edge.
3. Find the nearest one among the secondary primitives to be connected to this edge from the PACG.
4. If this is a corner, then the nearest point on the boundary edge is changed to the corner.
5. If this is an intersection or step edge, then project the boundary edge to the edge.
6. Repeat 2 to 5 until all the boundary edges will be refined.
7. Repeat 1 to 6 until all the patches will be refined.

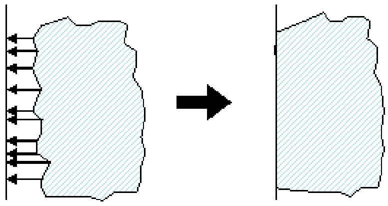

Figure 12. Refining roof patch with an edge

4.2 Wall Facets

Wall facets should be generated hypothetically because the vertical faces of an object are hardly observed from LIDAR data. Only the horizontal locations of the building outlines are accurately provided by the digital map. The step edges regardless of being derived from the building boundary of a map or from the patches connected in 2D indicates the existence of vertical facets of themselves. We thus derive a vertical patch between a pair of step edges to fill the gap between them with this patch. Figure 13 shows examples of the wall facets generated based on this method.

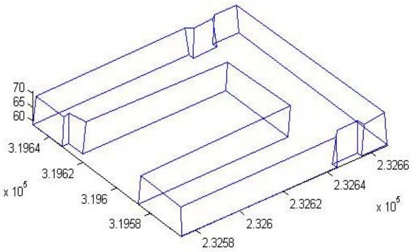

Figure 13. Wall facets

4.3 Polyhedral Model

The derived roof and wall facets are grouped into a polyhedral model. Among these facets, those connected to each other in particular share some edges and corners. Such redundant shared edges and corners are unified. Finally, the polyhedral model is examined with a topological test to check the completeness of the model. Inconsistency found among the facet, edges, and corners of a model indicates the existence of gaps in the model. These gaps are just identified so that they can be later edited if necessary. Figure 14 shows an example of the final polyhedral model (building 2).

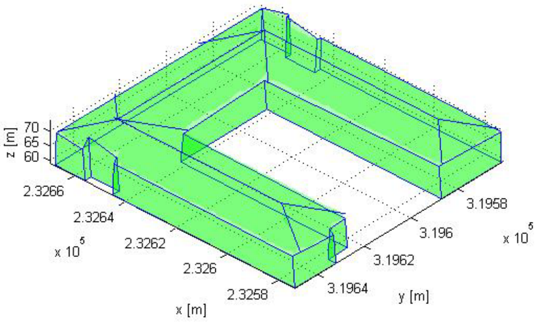

Figure 14. Polyhedral model

5. EXPERIMENTAL RESULTS

The proposed extraction approach was implemented as a program coded using C++ with standard template library. This program was applied to extracting building models from the input data of the test site mentioned in Section 2.2. The modeling results from the 13 buildings existing in this site are presented in Figure 15. They are the results from fully automatic processes without any manual intervention during this process or any manual editing after it. All the buildings presented from the map mainly retaining large and complex roof structures were verified to be reasonably modeled with visual inspection.

We inspected each building model based on its appearance in the aerial image, the LIDAR point plot, and the digital map. The inspection results of three buildings (ID: 4, 12, 10) are presented as follows.

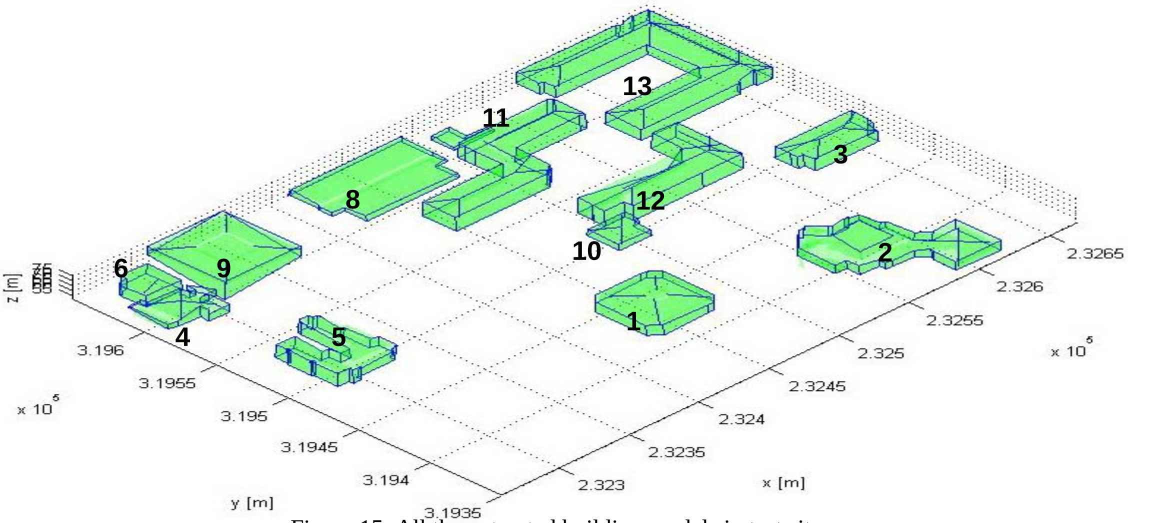

Figure 15. All the extracted building models in test site

Figure 16 shows the model of building 4 retaining the width of about 125 m, the length of about 56 m, and the height of 18 m. The generated model represents the winding of the wall around position A, which indicates the modeling process recovers narrow vertical facets.

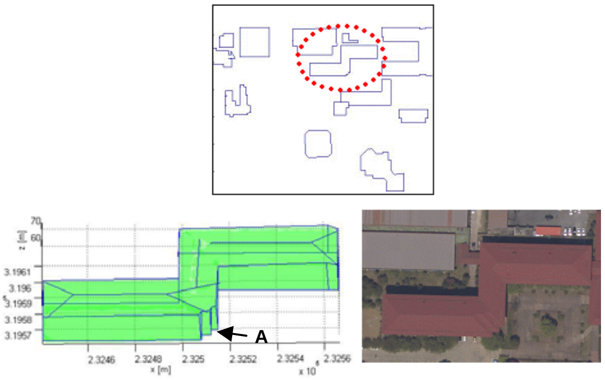

Figure 16. Extracted model of building (ID: 11)

The next building (ID: 2) is the most complicated shape. This building has many step edges and intersection edges on the roof. In addition, the ground plan is also irregular. Although the shape of the building is so peculiar that it might be difficult to apply a traditional model-based approach, the extracted model completely describes the shape in detail, as shown in Figure 17.

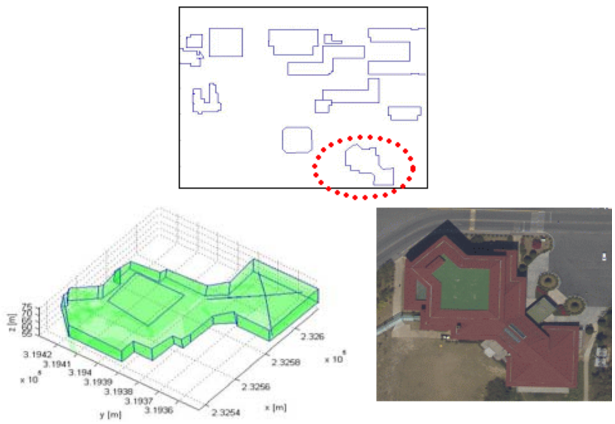

Figure 17. Extracted model of building (ID: 2)

Building 5 also has very complex roof structure in which many step edges are observed. The roof structure is modeled with 16 patches of various shape and size. The size of the smallest one is just 4 m^2, indicating how detail the proposed process can model a building.

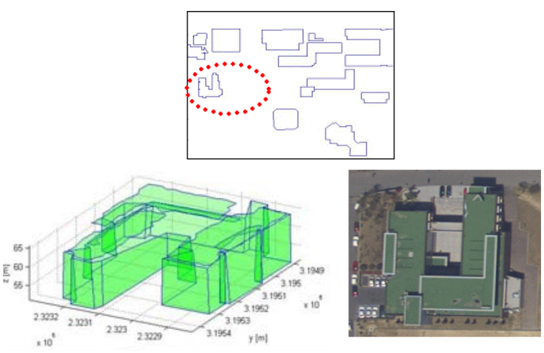

Figure 18. Extracted model of building (ID: 5)

6. CONCLUSION

We proposed an automatic method to extract three-dimensional detail models of buildings in particular with large-complex roof structure from LIDAR data and a digital map. From the modeling results of 13 buildings in the test site, the proposed method is verified to successfully extract the detail polyhedral models.

Most countries have already constructed large-scale maps including building layers. If they improve these maps to include 3D models of buildings for various applications such as 3D car navigation, the proposed method can be a time- and cost-effective solution.

Buildings newly built after a map being generated cannot be identified from the map. In order to model them in addition, we are studying a method to identify the existence of building and generate the step edges along the building only from LIDAR data.

REFERENCE

Baillard, C. and A. Zisserman, Automatic reconstruction of piecewise planar models from multiple views. In: *Proc. IEEE Conference on Computer Vision Pattern Recognition*, pp. 559–565.

Brenner, C. 2003. Building reconstruction from images and laser scanning. In: *Proc. ITC Workshop on Data Quality in Earth Observation Techniques*, Enschede, The Netherlands.

Brenner, C., 1998. Rapid acquisition of virtual reality city models from multiple data sources. In: *Int. Arch. Photogramm. Remote Sensing*, Vol. 32, Part 5, pp. 323-330.

Edelsbrunner, H., D. G. Kirkpatrick, and R. Seidel, 1983. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, Vol. 29, No. 4, pp. 551-559.

Lee, I., 2002. *Perceptual Organization of Surfaces*, Ph. D. Dissertation, The Ohio State University, Columbus, Ohio, USA.

Lodha S. K., K. Kumar, and A. Kuma, 2005. Semi-automatic roof reconstruction from aerial LIDAR data using K-means with refined seeding, In: *ASPRS Conference*, Baltimore, Maryland.

Moon J., I. Lee, S. Kim, and K. Kim, 2005. Outlier Detection from LIDAR Data based on the Point Density. *KSCE journal*, 25(6D), pp. 891-897.

Rottensteiner, F and C. Briese, 2003. Automatic Generation of Building Models from LIDAR Data and the Integration of aerial image, In: *Int. Arch. Photogramm. Remote Sensing*, Vol. 34.

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

Vosselman, G. and Dijkman, 2001. 3D building model reconstruction from point clouds and ground plan, In: *Proceedings of the ISPRS Workshop on Land Surface Mapping* *and Characterization Using Laser Altimetry*, Annapolis, MD, pp. 37-44.