Remote Sens. **2013**, *5*, 5944-5968; doi:10.3390/rs5115944

OPEN ACCESS

Remote Sensing

ISSN 2072-4292

www.mdpi.com/journal/remotesensing

Article

Knowledge-Based Modeling of Buildings in Dense Urban Areas by Combining Airborne LiDAR Data and Aerial Images

Junichi Susaki

Graduate School of Engineering, Kyoto University, Kyotodaigaku Katsura, Nishikyo-ku,
Kyoto 615-8540, Japan; E-Mail: susaki.junichi.3r@kyoto-u.ac.jp;
Tel.: +81-75-383-3300; Fax: +81-75-383-3300

Received: 11 July 2013; in revised form: 28 October 2013 / Accepted: 5 November 2013 / Published: 14 November 2013

Abstract: In this paper, a knowledge-based algorithm is proposed for automatically generating three-dimensional (3D) building models in dense urban areas by using airborne light detection and ranging (LiDAR) data and aerial images. Automatic 3D building modeling using LiDAR is challenging in dense urban areas, in which houses are typically located close to each other and their heights are similar. This makes it difficult to separate point clouds into individual buildings. A combination of airborne LiDAR and aerial images can be an effective approach to resolve this issue. Information about individual building boundaries, derived by segmentation of images, can be utilized for modeling. However, shadows cast by adjacent buildings cause segmentation errors. The algorithm proposed in this paper uses an improved segmentation algorithm (Susaki, J. 2012.) that functions even for shadowed buildings. In addition, the proposed algorithm uses assumptions about the geometry of building arrangement to calculate normal vectors to candidate roof segments. By considering the segmented regions and the normals, models of four common roof types—gable-roof, hip-roof, flat-roof, and slant-roof buildings—are generated. The proposed algorithm was applied to two areas of Higashiyama ward, Kyoto, Japan, and the modeling was successful even in dense urban areas.

Keywords: 3D building modeling; dense urban areas; airborne LiDAR; aerial image; image segmentation

1. Introduction

Disaster preparedness and mitigation require accurate information about buildings and other infrastructure that might be impacted by a typhoon, earthquake, or other disaster. For example, Kyoto, the ancient capital of Japan, is a UNESCO World Heritage site with hundreds of irreplaceable structures, many of which are wooden. The active Hanaore fault lies under downtown Kyoto. If an earthquake occurs, fire may spread through dense urban areas and the valuable historical structures, located in the dense urban areas, would be burnt and lost. Architects and engineering researchers have attempted to simulate the damage caused by fire to support evacuation planning and other disaster prevention measures. A physics-based fire spread model has been developed [1], and its performance examined by application to the dense urban areas of Kyoto [2]. Such models require many building parameters including three-dimensional (3D) building models. However, 3D building models are not available to researchers at a reasonable cost. Google Earth provides 3D building models, but its models of the Higashiyama area are poor because dozens of Japanese houses are modeled as a single box model. Better simulation of fire spread requires a model that separates individual buildings [1].

Airborne light detection and ranging (LiDAR) can provide the data necessary for generating three-dimensional models. 3D building modeling using airborne LiDAR has been studied for more than a decade, and existing algorithms have produced both simple models [3–4] and more detailed models [5–7]. Building modeling is principally based on the segmentation of point clouds into a number of point groups that correspond to each building. The degree of detail in the model depends on the requirements of the model and the density of the LiDAR data. The point groups obtained during the process are used in the model to represent the shape, height, and spatial arrangement of the real buildings. This method assumes that the original LiDAR point clouds can be classified into small groups of points that reflect the individual buildings.

Automatic 3D building modeling in dense urban areas is a well-known challenge. For example, in dense urban areas in Kyoto, Japan, traditional Japanese houses are located close to each other, and the heights of the ridges on the roofs are similar. Modeling of buildings in some urban areas might be possible using current techniques, even if the buildings are located close to one another, as their heights are often dissimilar and differences in height can be easily detected. In contrast, houses having similar heights might be difficult to model using existing algorithms because the point cloud cannot be separated into groups representing individual houses based on height information alone.

To improve modeling accuracy, some researchers have examined fusion of airborne LiDAR data with a digital map [8]. As such approaches require a pre-existing digital building map, they might be costly to apply to large urban areas. As alternatives to digital maps, other data sources have been examined, such as aerial or satellite images [9–13] and terrestrial LiDAR data [14]. In particular, aerial images are promising for providing data showing individual buildings in dense urban areas. Existing research shows that a key aspect of efficient data fusion in 3D modeling is obtaining building geolocation and shape information from 2D images, in addition to obtaining 3D coordinate data from airborne LiDAR data. However, to the author’s knowledge, even when combining data sources, no successful modeling algorithm has been reported that can be applied to an area where houses of similar height are located close to one another. The main reason for this shortcoming may be due to the difficulty in segmentation of buildings located close to each other from images.

Object-based segmentation is a conventional approach for extracting building geolocation and shape information from images. This segmentation technique is widely used in a number of applications, including the mapping of urban areas [15–17]. However, in our previous research [18], object-based segmentation using commercial software (ENVI EX Version 4.8) produced poor results when tested on an image of dense urban areas with spatial resolution of 25 cm. Factors in the unsuccessful segmentation were as follows: (1) unclear building boundaries caused by roofs with similar brightness being located close to each other; (2) shadows cast by neighboring buildings; and (3) roofs having a rough texture. Our paper cited above proposed an algorithm that considers these factors and that can effectively segment buildings, including shadowed buildings, in dense urban areas through the use of aerial images.

In the current paper, another algorithm is proposed for automatic generation of 3D building models for dense urban areas, by using the building segmentation results from aerial images produced by the aforementioned algorithm in combination with airborne LiDAR data. In addition to information about the building area, normals to the roofs are determined for groups of buildings that are linearly aligned. By taking account of the segmented regions and the normals, the method selects the correct building roof types and generates models that are more accurate than those produced by previous methods.

The remainder of this paper is organized as follows. An overview of our previously reported segmentation algorithm is given in Section 2, along with an explanation of the newly proposed method. The locations used in the study, and the data collected from these sites are described, and the experimental results are reported in Section 3. The implications of these results and the validity of the algorithm are then discussed in Section 4. Finally, Section 5 concludes the paper.

2. Methodology

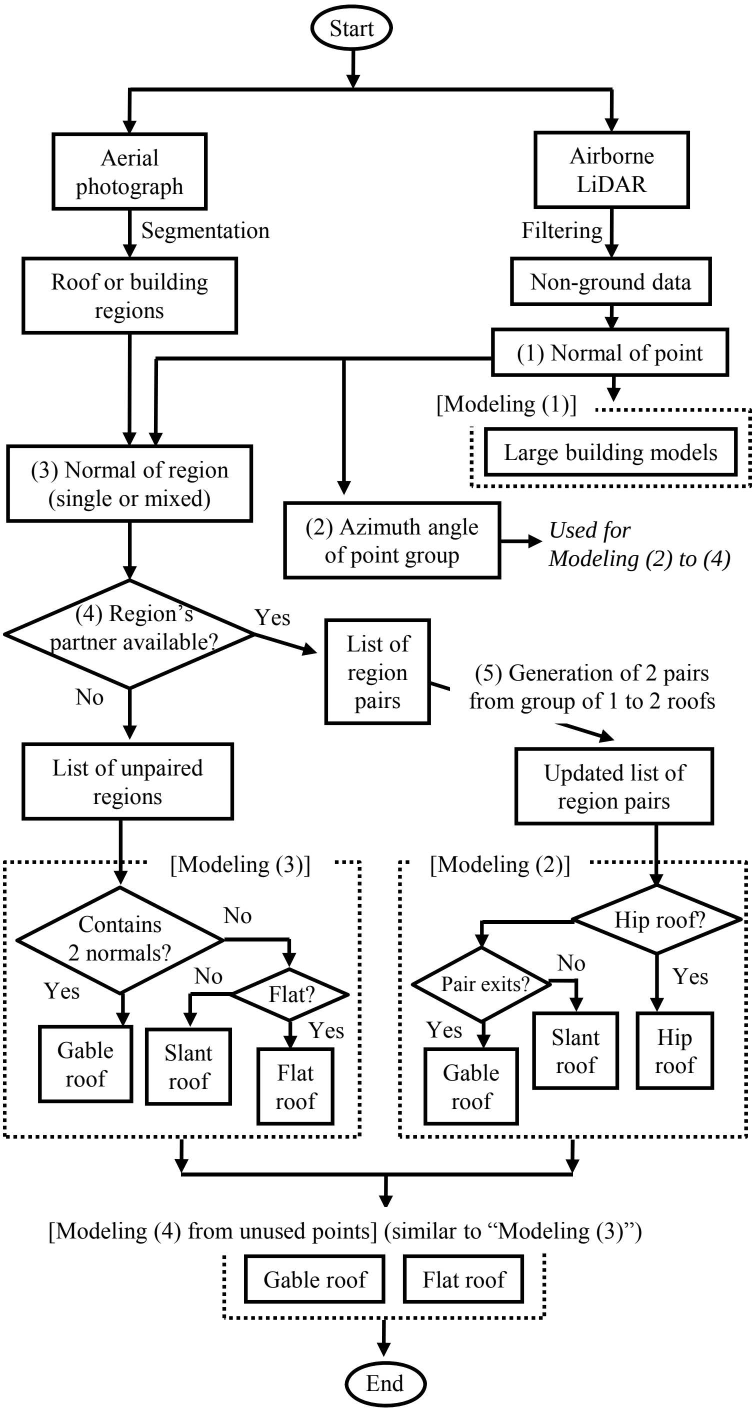
The proposed algorithm for automatically generating 3D building models consists of three parts: (1) image segmentation, (2) LiDAR data filtering, and (3) 3D building modeling. Figure 1 shows a flowchart of the algorithm. The algorithm assumes four simple building types: gable-roof, hip-roof, flat-roof, and slant-roof buildings. Two types of hip-roofs are included: two-sided and one-sided (Figure 2). The segmentation algorithm [18] and filtering algorithm [19] used in the current work have been described elsewhere, and the details of the algorithms are omitted in this paper.

The 3D building-modeling algorithm introduced here first calculates region normals to generate models that are orderly arranged in dense urban areas. Then, incomplete segmented results are improved before modeling. Finally, the algorithm generates 3D models under four modeling strategies. In cases where models generated overlap each other, model boundaries are corrected. The details of the algorithm are described in the following subsections.

2.1. Calculation of Region Normals

In preliminary modeling of buildings in dense urban areas, we observed that the azimuthal direction of the roof normal is important to generate models in dense urban areas. However, if normals are estimated on a per-building basis, the spatial arrangement of generated building models will be independent, and then they will be difficult to keep in a line, a common pattern found in dense urban areas. The proposed algorithm for generating 3D building models focuses on the geospatial arrangement of buildings in dense urban areas. Principal azimuthal directions of roofs are estimated based on building groups and utilized for determining individual roof normals (Figure 3).

Figure 1. Flowchart of proposed algorithm to generate building models.



In Step (1), the algorithm calculates the normal of each LiDAR point using neighboring points within a window (Parameters (a) and (b) of Table 1) by fitting a plane. If the root mean square error (RMSE) is below a threshold (Parameter (c)), the obtained normal is used for all points within the window. Here, the vertical component of the normal (n_x, n_y, n_z) is defined as n_z, where $n_x^2 + n_y^2 + n_z^2 = 1$. If the vertical component of the normal is greater than another threshold (Parameter (d)), the normal is labeled as a "vertical normal"; otherwise, the normal is labeled with a direction (north, south, east, or west).

Figure 2. Building models generated by the proposed algorithm: (**a**) gable-roof, (**b**) hip-roof, (**c**) flat-roof, and (**d**) slant-roof building model.

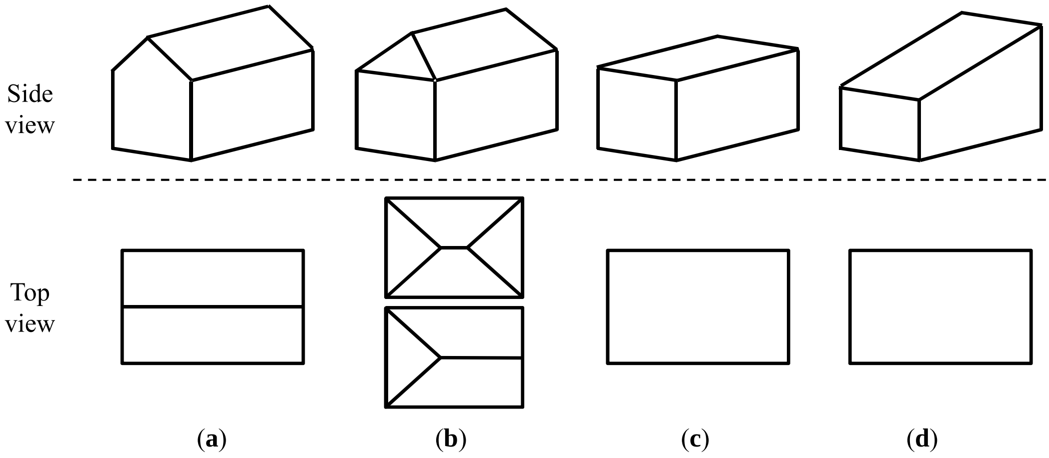
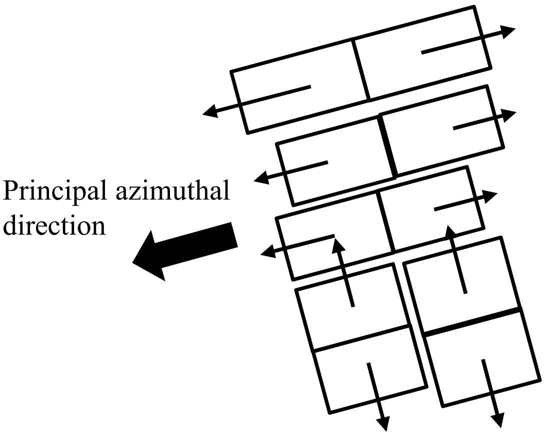


Figure 3. Determination of principal azimuthal direction of a point cloud.



Step (2) segments the point clouds into groups on a building-block basis by examining the horizontal and vertical distances to each point's nearest neighbors. If these distances are less than the respective thresholds (Parameters (e) and (f)) the point is added to the building block containing the neighbor. Each group of points that form a building block is allocated a principal azimuthal direction by finding the peak of the probability distribution generated from the normals to the points (Figure 4). The definition of the probability is the ratio between the frequency of a specific angle (degree) and the sum within the building block of frequencies in all angles (360 degrees). These azimuthal directions are used in Modeling (2) to (4).

2.2. Improvement of Incomplete Information from Image Segmentation

Segmented results can help in modeling, but those results will likely contain errors. Some roofs may be missing, some roof planes belonging to the same building may be unified, or some roofs belonging to neighboring buildings may be falsely segmented together. In this subsection, we describe techniques used to improve the quality of the segmented results.

Table 1. Experimental parameter values.

Process	Parameters	Value used	Type
Step (1): Normal calculation	(a) Window size	4.5 m × 4.5 m	2
	(b) Minimum data number to calculate	6 points	4
	(c) Maximum root mean square error (RMSE)	0.1	4
	(d) Minimum vertical component of normal for flat roof	0.995	3
Step (2): Determination of azimuth angle of points	(e) Horizontal and vertical distance for segmentation of point clouds	3.0 m	1
	(f) Minimum height for discrimination of roof points	2.0 m	1
Step (3): Determination of normal of regions	(g) Minimum possibility for single normal class	0.7	3
	(h) Minimum possibility for two normal classes	0.8	3
Step (4): Matching of regions	(i) Minimum valid length	1 m	4
	(j) Maximum invalid length	1 m	4
	(k) Minimum ratio between maximum length of the smaller region and that of the larger region	0.25	1
Determination and correction of boundaries of models	(l) Ratio r for boundary search	0.1	4
Modeling (1): Large building	(m) Horizontal and vertical distance for segmentation of point clouds (used also in Modeling (4))	2.0 m	1
	(n) Minimum number of points (used also in Modeling (4))	50 points	3
Modeling (2)	(o) Minimum length of rectangle (used also in Modeling (3) and (4))	2.0 m	1
	(p) Minimum area (used also in Modeling (3) and (4))	8.0 m^2	1

Notes: "Type" represents factors affecting the selection of the parameter. "1" denotes artificial geographical features (e.g., average building area/size, and street width), "2" denotes LiDAR point density, "3" denotes both artificial geographical features and LiDAR point density, and "4" denotes others.

Step (3) classifies the building regions into different normal types based on the normals calculated in Step (1). Figure 4 shows that both Regions α and β have a single normal direction, whereas Region γ has two major normals. For each region, a probability distribution of the five normal directions of the points (vertical and the four cardinal directions) included in the region is generated. If the maximum probability is greater than a given threshold (Parameter (g)), the region is assigned a single normal class and the regional "rough direction" is determined. The exact normal is calculated by selecting the normal that is closest to the regional rough direction. If the maximum probability is less than the threshold and the sum of the two highest probabilities is greater than another threshold (Parameter (h)), the region is given a "mixed normal" classification. In addition, the algorithm examines whether the azimuthal directions of the two highest probabilities are linearly symmetric to each other (e.g., the relationship between Regions α and β of Figure 4). If they are linearly symmetric, the region is divided into two gable roofs (Regions γ_1 and γ_2 of Figure 4).

Step (4) searches for a "normal pair" partner for a region that has a single normal and that has not been assigned a "vertical class." We conclude that we have found a partner when the candidate partner region has a single normal class with a normal direction that is linearly symmetric to the original normal direction, or when the partner region has a mixed normal class. The search is along the direction opposite to the azimuthal direction of the region normal. If another region is detected and a pixel length having a non-null value exceeds a threshold value (Parameter (i)), the search terminates. The search also terminates if no region is detected for which an invalid length exceeds another threshold (Parameter (j)). First, however, the algorithm computes the ratio between the maximum length of the smaller region and that of the larger region, and if that ratio is less than a threshold (Parameter (k)), the partner is rejected.

Step (5) uses real-world spatial knowledge to further improve the identification of partner regions. This step assumes, given a certain pattern of results, that two gable-roof buildings are available in reality, but that the four roof areas of these two buildings are not properly extracted by segmentation of the aerial image. An example is shown in the left part of Figure 5. These regions are denoted “groups of 1 to 2 roofs” hereinafter. Two pairs of roofs are generated by dividing Region γ into two regions, Region α′ and Region β′ (Figure 5 (right)).

Figure 4. Determination of principal azimuth of a region. “N”, “E”, “S”, and “W” denote north, east, south, and west, respectively. As the two dominant normals of Region γ are linearly symmetric, the region is divided into two gable roofs, Regions γ_1 and γ_2.

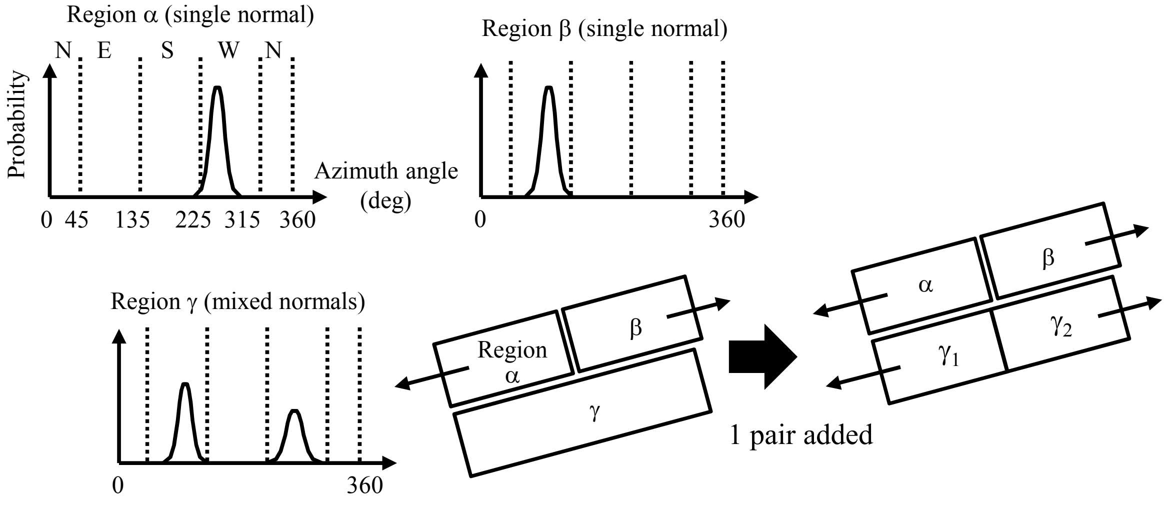
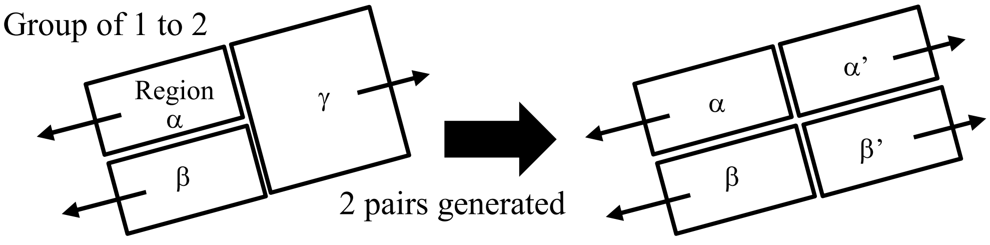


Figure 5. Generation of two gable-roof buildings when the four roofs are not correctly extracted. These regions are called a group of 1 to 2 roofs in this paper. Region γ is divided into two regions, Region α′ and Region β′.



2.3. Multi-Strategy Modeling

The proposed algorithm includes four modeling strategies. First, Modeling (1) is conducted after Step (1), with the aim of modeling relatively large flat-roof buildings. In segmenting the aerial image, we found that commercial buildings, parking structures, and other flat-roof buildings were often segmented into small areas because of objects on the roof, making it inappropriate to model such buildings using the segmented results. Therefore, Modeling (1) is conducted using only the LiDAR data without the segmented results. Point clouds are segmented by setting vertical and horizontal distance thresholds (Parameter (m)). Point clouds having a certain number of points (Parameter (n)), and whose dominant normal class is "vertical", are used to generate large flat-roof building models.

As a result of improving incomplete information, two lists are generated: a list of region pairs, and a list of regions without pairs. Modeling (2) uses the first list. It generates models for those regions whose normals are not vertical and whose partners are available, in descending order of each region's area. A point shared by two regions is determined to be a point on a ridgeline. If no point is shared by two regions, a centroid of edges of the two regions is used as an alternative. The ridge direction is established such that it is orthogonal to the azimuthal direction of the two regions' normals. A search is conducted, on both sides along the ridge direction, for regions whose normal azimuthal direction is parallel to the ridge's direction. If two such regions are detected, a two-sided hip-roof building model is generated. If one such region is detected, a single-sided hip-roof building model is generated. If no such region is detected, a gable-roof building model is generated. During this modeling process, the minimum length of the rectangle (Parameter (o)) and minimum area (Parameter (p)) are checked to exclude small models that are probably artifacts.

Using the list of regions without pairs, Modeling (3) generates models for the regions remaining after Modeling (2). If a region has a mixed normal class and its two azimuthal directions are linearly symmetric (e.g., Region γ in Figure 4), a gable-roof building model is generated. If the region has a single slanted normal, a slant-roof building model is generated. Otherwise, a flat-roof building model is generated.

After Modeling (3), some points may remain unused because they did not meet the criteria for strategies (1) through (3). In addition, there may be inconsistencies between the LiDAR data and aerial image observations. When a building does not exist in the aerial image, but does exist in the LiDAR data, the area of the building will not be segmented. In this case also, the points of the building area will remain unused. Modeling (4) generates models using the unused points. Similarly to Modeling (3), additional gable-roof or flat-roof building models are generated based on the azimuthal direction. If some points still remain unused after this process, flat-roof building models are generated without using the azimuthal direction (Parameters (m) and (n)). Because points used in Modeling (4) may include points reflected from tall vegetation, vegetation removal should be executed before modeling. This issue will be discussed in Section 4.1.

2.4. Determination and Correction of Model Boundaries

In the proposed algorithm, the boundaries of each building are assumed to be orthogonal or parallel to the principal azimuthal direction of its associated group of buildings. That is, buildings are assumed to be rectangular. The boundaries of a modeled building are generated such that their directions are parallel or orthogonal to the principal azimuthal direction of the building-group point cloud in which the model is included. In actuality, a model's geolocation is highly dependent on the regions segmented using the aerial image. Because geolocational and sizing errors could be included in the segmented regions, a number of corrections are performed, as shown in Figure 6. We determine the "true" location of an edge by doing a local search and evaluating edges along the principal direction of a region and the orthogonal direction. Edges of segmented regions from aerial image are projected onto the principal azimuthal direction of a region (called the "u-axis"), and the probability density is generated. The definition of the probability $P(u)$ is:

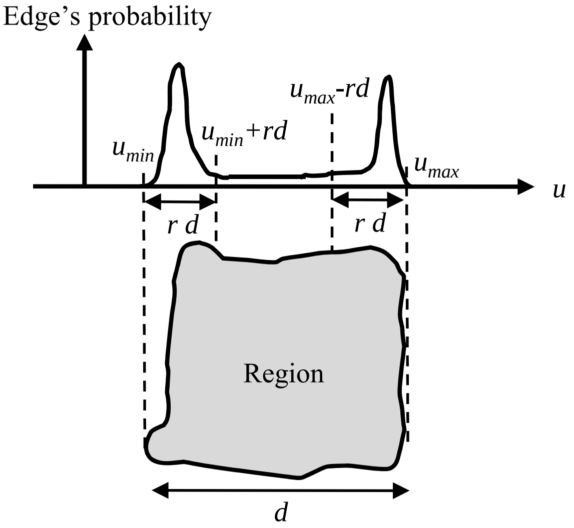
$$P(u) = n(u)/N \tag{1}$$

where $n(u)$ denotes edge frequency at a specific u and N denotes the total number of edges in the region. The width d of the region along the u-axis is defined as:

$$d = u_{max} - u_{min} + 1 \tag{2}$$

where u_{max} and u_{min} denote the maximum and minimum u values, respectively. The maximum probability is searched for within an interval of width d multiplied by a prescribed ratio r (Parameter (l)), that is, $[u_{min}, u_{min} + rd]$ and $[u_{max} - rd, u_{max}]$ for lower and upper boundary detections, respectively. The u value that occurs with highest probability among the two intervals is selected as the boundary location. The same process is repeated along an axis orthogonal to the principal direction (called the "v-axis").

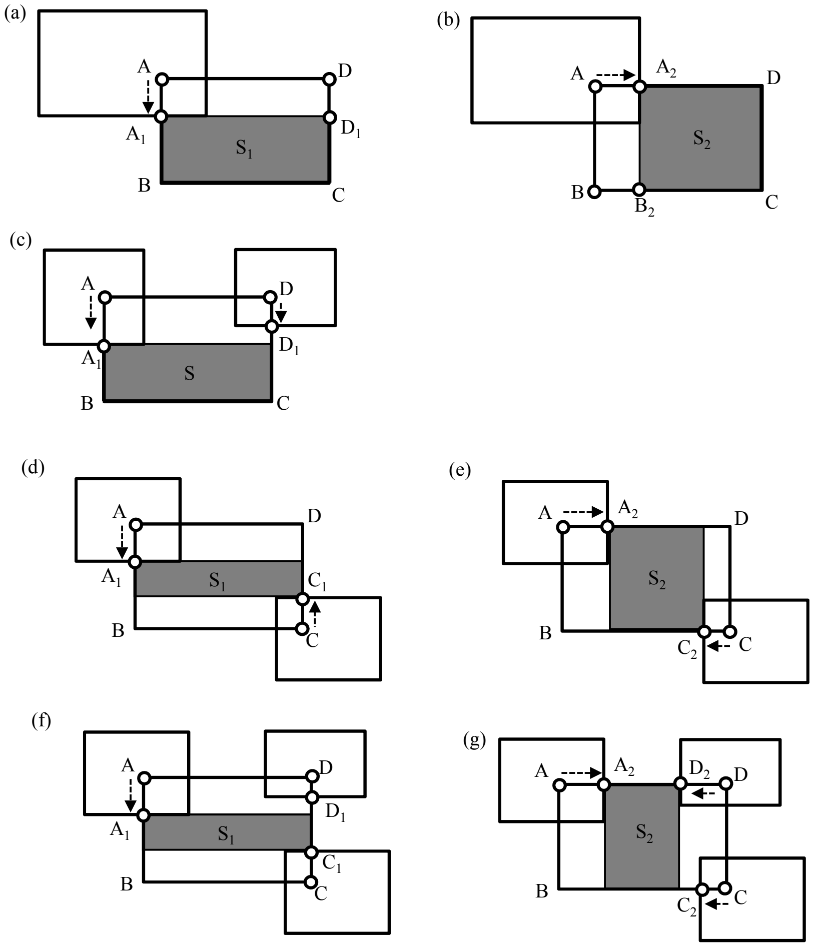
Figure 6. Delineation of the rectangular boundary of the model. Edges are projected onto the principal direction of the region (called the "u-axis"), and a probability distribution is generated. Maximum probability is searched within an interval of width *rd*, and the boundary is determined at the u value whose probability is maximum.



If a generated model overlaps an existing model, the boundary of the model is corrected by implementing the following process. If one vertex of a rectangle overlaps the area of another modeled building, as shown in Figure 7a,b, the overlapping vertex (Point A) is shifted in two directions (toward

Points A_1 and A_2) along the rectangle's sides. Accordingly, the sides connected to the vertex are also shifted (AD to A_1D_1 and AB to A_2B_2). The shifted vertex is selected on the model whose rectangular area, S_1 or S_2, is the greater.

Figure 7. Correction of a model's boundary when a generated model overlaps an existing model: (**a**,**b**) if a vertex of a rectangle overlaps one of the four vertices, the overlapping vertex (Point A) is shifted in two directions (toward Points A_1 and A_2) along the rectangle's sides. Accordingly, the sides connected to the vertex are also shifted (AD to A_1D_1 and AB to A_2B_2). The shifted vertex is selected from the rectangle with greater area; (**c**) if two adjacent vertices overlap, the length A_1B and CD_1 are compared and the shorter is selected for relocating the side; (**d**,**e**) if two diagonal vertices overlap, the shifted vertices are selected on the model whose rectangular area, S_1 or S_2, is the greater; and (**f**,**g**) if three vertices overlap, a procedure is executed that combines the procedures for two adjacent and two diagonal vertices overlapping. The shifted vertices are selected on the model whose rectangular area, S_1 or S_2, is the greater.



If two adjacent vertices overlap another building, as shown in Figure 7c, the side between the vertices is shifted to avoid overlapping. In Figure 7c, lengths A_1B and CD_1 are compared and the shorter is selected for relocating the side. If two diagonal vertices overlap, as shown in Figure 7d,e, a procedure similar to that for single vertex overlapping is performed. The shifted vertices are selected on the model whose rectangular area, S_1 or S_2, is the greater. If three vertices overlap, as shown in

Figure 7f,g, a procedure is executed that combines the procedures for two adjacent and two diagonal vertices overlapping. The shifted vertices are again selected on the model whose rectangular area, S_1 or S_2, is the greater.

3. Experiments

3.1. Study Area and Data Used

Kyoto is famous for its many traditional houses and landscapes. Its complexity makes it a challenging target for 3D modeling. This research used areas in the Higashiyama ward of Kyoto. The Higashiyama ward is hilly and is well-known for its numerous temples and shrines, and for its narrow streets—approximately 5 to 6 m in width. Orthographically projected aerial images with a 25 cm spatial resolution were obtained for this research on 17 November 2007.

Details of the airborne LiDAR data are shown in Table 2. Note that the LiDAR data were acquired nearly five years before the aerial imagery. The LiDAR density was calculated by projecting the point cloud onto the aerial image, noting which pixels had corresponding points, and finding the ratio between the number of these pixels and the total pixels for the area. In the case of a 1 m cell size, it was found to be approximately 0.7 points/m^2. The effect on the final modeling of the difference in acquisition time between the two primary data sets is discussed in Section 4.1.

Table 2. Details of airborne light detection and ranging (LiDAR) used for this study.

Measurement date	June 2002 to February 2003
Measurer	Aero Asahi Corporation
Density (calculated as valid pixels/all pixels in the area)	0.70
Altitude	900 to 1,000 m
Horizontal accuracy	±50 cm
Vertical accuracy	±15 cm

3.2. Results

The source code of the proposed algorithm was developed by the author in C and executed in the Cygwin environment on Windows 7 using a PC with an Intel Core i7 (3.20 GHz) processor and 6 GB of memory. Experiment parameters were set to the values shown in the right-hand column of Table 1. These parameter values were determined empirically during initial experimentation. The parameters for Steps (1) and (2), and Modeling (1) to (4) depend on building size, road width, and LiDAR data density. Parameters for Step (3) depend on dominant building types. Parameters for Step (4) depend on aerial image resolution and building size. Note that "Horizontal and vertical distance for segmentation of point clouds" is larger in Step (2) than in Modeling (1) and (4) because Step (2) segments point clouds into building block groups instead of individual buildings.

Although the slant angles of individual roofs can be determined by using airborne LiDAR data, we found that some of the estimated slant angles were unstable, partly because small roofs may not contain enough data points to calculate their normal, owing to occlusion, and partly because the points belonging to neighboring buildings may be included when calculating normals. Many of the gable roofs and hip roofs in the study area have a similar slant angle and, therefore, minimum and maximum slant angles (17 degrees and 45 degrees, respectively) were set for the research. Ridges, which were assumed to be horizontal, were estimated by calculating the crossing line of two roofs.

To reduce the computation time, we ran our algorithm on 125 m × 125 m subareas, and then merged the results. Due to this subdivision of the study area, buildings that cross the boundaries of subregions may be modeled as smaller than the actual building. Examining the detailed results of the experiment, we did not find any building that displayed this problem. Figure 8 shows an aerial image, and Figure 9 shows the segmented result obtained by using the algorithm proposed previously [18]. Airborne LiDAR data were filtered by using our previously proposed algorithm [19]. Five classes of normals of this area, vertical and in the four cardinal directions, were generated, and a 5 × 5 window was applied to select the major normal direction. Figure 10a–d show wireframe models generated in Modeling (1) to Modeling (4). All of the models are merged in Figure 10e. In the models, red denotes a flat-roof building, blue denotes a gable-roof building, green denotes a hip-roof building, pink denotes a slant-roof building, black denotes a flat big building, and yellow denotes a flat model generated using finally unused points. In the same manner, the proposed algorithm was applied to other areas: areas containing low-rise buildings, and an area containing low-rise buildings and high-rise and complex buildings. The results are shown in Figures 11, 12 and 13, respectively.

Figure 8. Airborne LiDAR data (**left**) and aerial image (**right**).

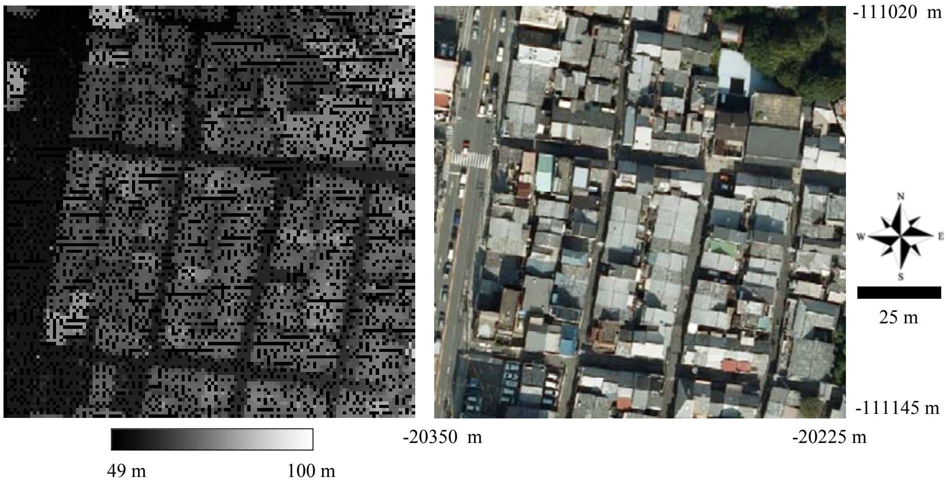


Figure 9. Segmentation result for Figure 8 by the algorithm proposed in [18].

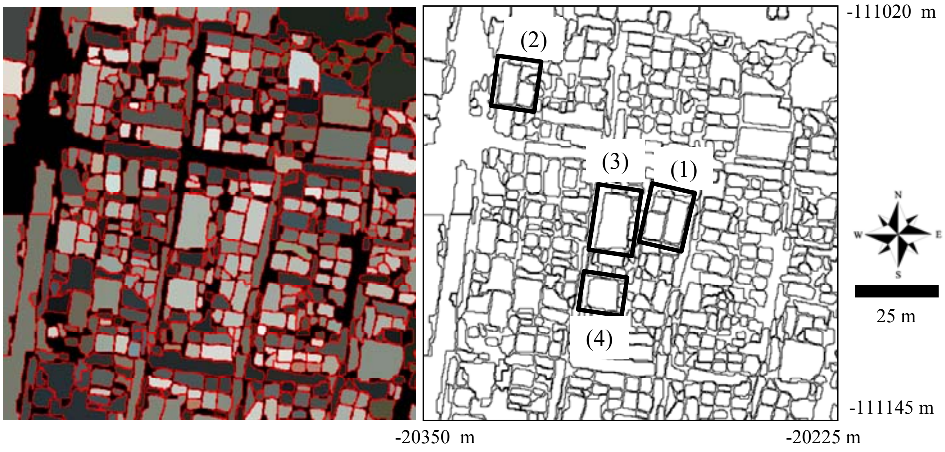


Figure 10. Wireframe models generated by the proposed algorithm on the area of Figure 8. (**a**–**d**) show models in "Modeling (1)" to "Modeling (4)" of Figure 1, and (**e**) shows models of (**a**) to (**d**). In the models, red denotes a flat-roof building, blue denotes a gable-roof building, green denotes a hip-roof building, pink denotes a slant-roof building, black denotes a flat big building and yellow denotes a flat model generated using finally unused points.

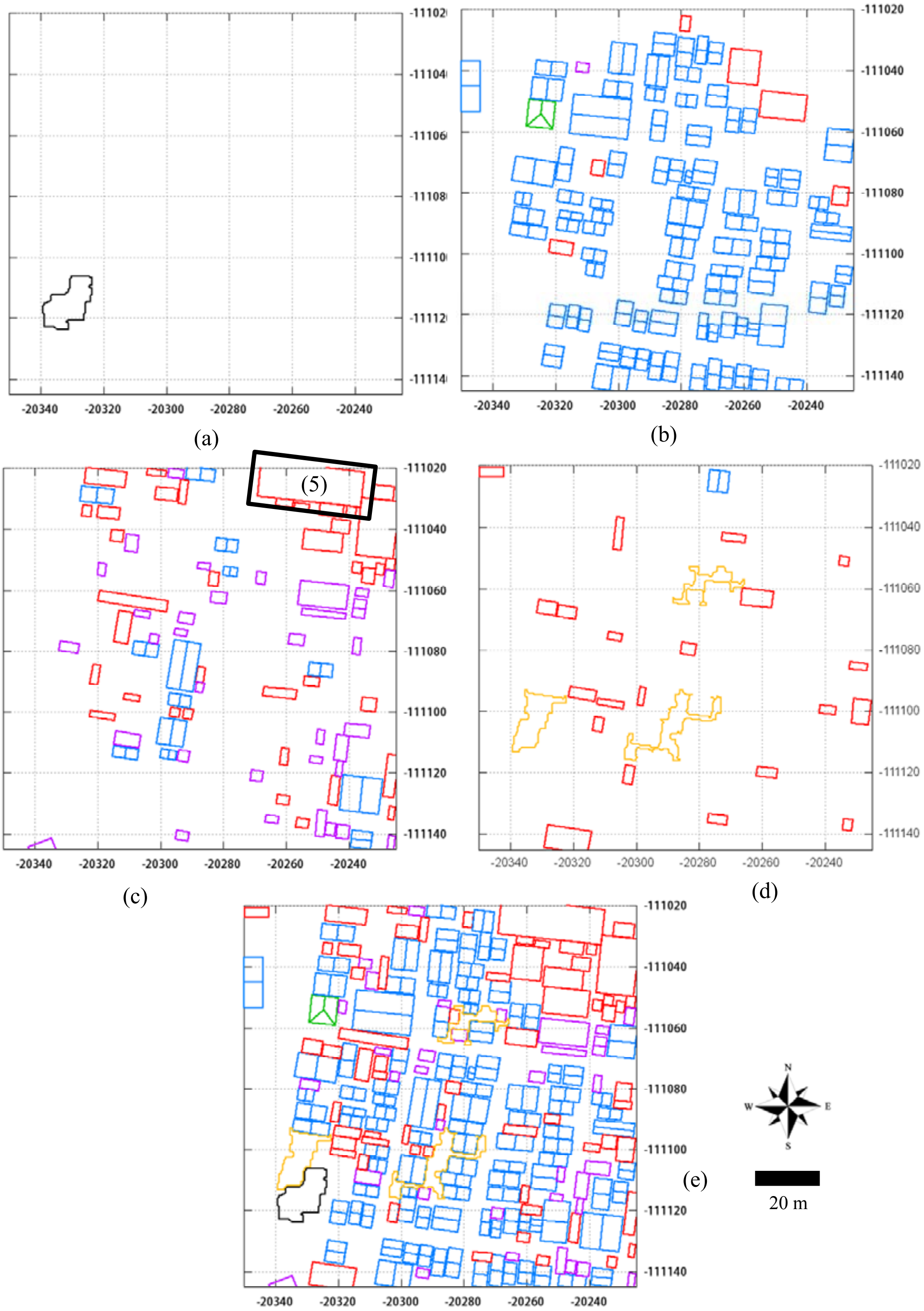
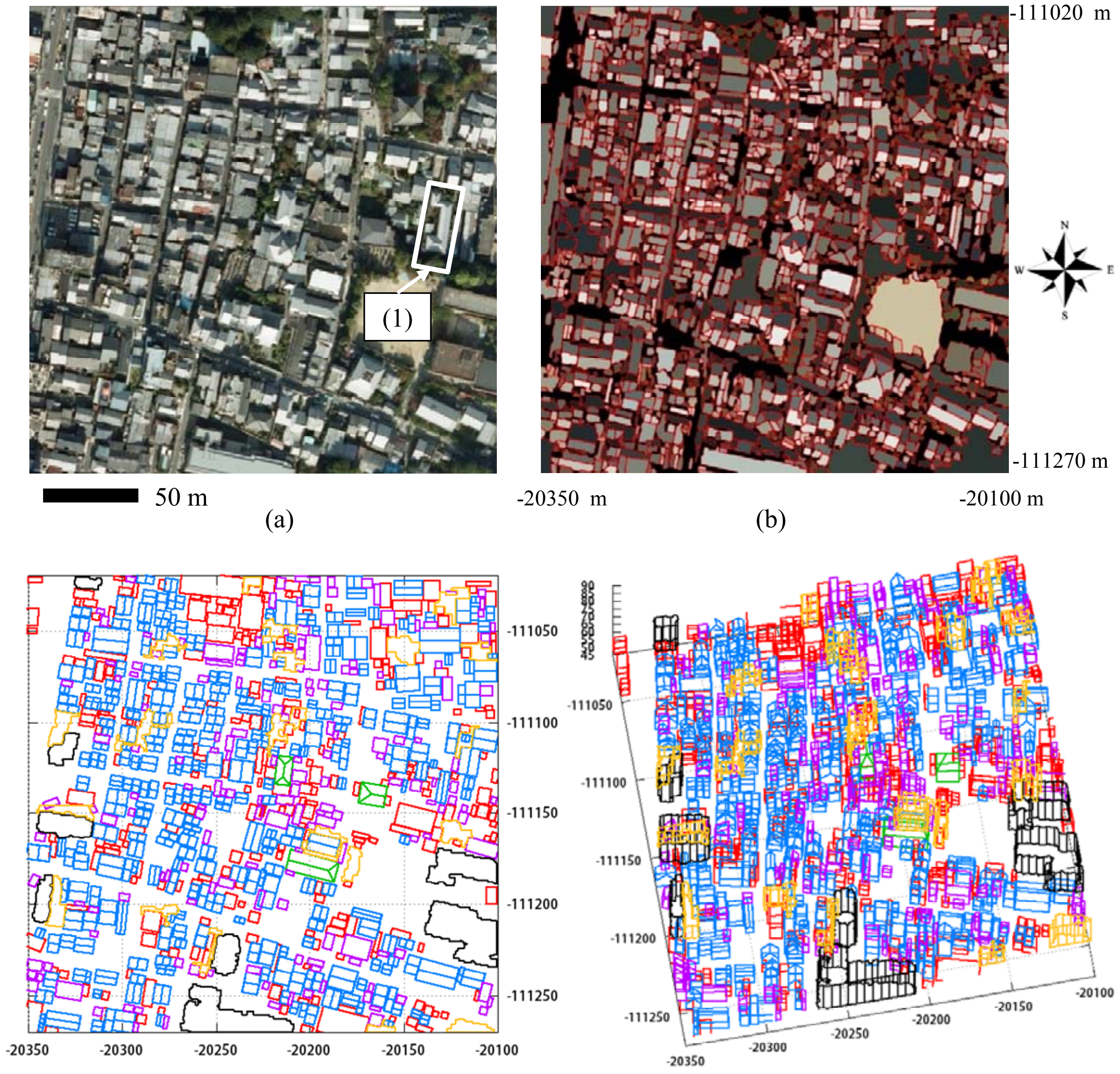


Figure 11. (**a**) Aerial image, (**b**) segmentation results, and (**c**,**d**) modeling results for an area containing low-rise buildings. Wireframe models are as in Figure 10.



(a) (b)

(c) (d)

3.3. Validation

The proposed modeling algorithm should be compared with other existing algorithms or software. However, no modeling software was available to the author. Instead, generated building models were assessed for the study area of Figure 11 in terms of type of building, the number of buildings, and the ground area of the models. The models were first validated not only by comparing the number of modeled buildings with the actual number of buildings (Table 3), but also by assessing the segmented results (Table 4). Although present in the study area, L-shaped gable-roof buildings and complex buildings were excluded from the analysis, because their numbers were limited (19 out of 382 buildings) and because our algorithm is not designed to handle them at this point. Therefore, only gable-roof, hip-roof, flat-roof, and slant-roof buildings were considered in the validation of the algorithm. If only part of a building was included in the aerial image and, therefore, the type of the generated model was incorrect due to the lack of information, the building was also excluded from the validation.

Table 3. Comparison between the buildings contained in the aerial image and those produced by the algorithm.

		Model						
		Gable	**Hip**	**Flat**	**Slant**	**Failure**	**Sum**	**Producer's Accuracy**
Actual	Gable	239	0	24	32	3	298	80.2%
	Hip	9	3	3	2	0	17	17.7%
	Flat	7	0	32	2	3	44	72.7%
	Slant	2	0	0	2	0	4	50.0%
	Sum	257	3	59	38	6	363	
	User's Accuracy	93.0%	100%	54.2%	5.3%			76.0%

Notes: Values in the table denote the number of corresponding buildings. || || denotes the number of entities, TP denotes "true positive," which are "objects" in the reference correctly classified as "objects", FN denotes "false negative," which are "objects" in the reference falsely classified as "background," and FP denotes "false positive", which are "background" in the reference falsely classified as "objects." Completeness = ||TP||/(||TP|| + ||FN||) = 76.0%, Correctness = ||TP||/(||TP|| + ||FP||) = 77.3%

Table 4. Comparison between segmentation results and the model landscape. Segmentation results were classified into five categories: (**a**) each roof successfully segmented; (**b**) all roofs successfully extracted, but roofs of the same building merged; (**c**) all roofs successfully extracted, but roof(s) merged with another building's roof(s); (**d**) not all roofs extracted and (**e**) no roof extracted. Values in the table denote the number of corresponding buildings.

			Model						
			Gable	**Hip**	**Flat**	**Slant**	**Failure**	**Sum**	**Producer's Accuracy**
Segmented results	Gable	(a)	137	0	7	12	2	158	
		(b)	19	0	4	7	1	31	
		(c)	71	0	8	3	0	82	
		(d)	11	0	5	8	0	24	
		(e)	1	0	0	2	0	3	
		Sum	239	0	24	32	3		80.2%
	Hip	(a)	7	3	1	2	0	13	
		(b)	1	0	1	0	0	2	
		(c)	0	0	0	0	0	0	
		(d)	1	0	0	0	0	1	
		(e)	0	0	1	0	0	1	
		Sum	9	3	3	2	0	17	17.6%
	Flat	(a)	6	0	19	2	2	29	
		(b)	0	0	0	0	0	0	
		(c)	0	0	1	0	0	1	
		(d)	1	0	3	0	1	5	
		(e)	0	0	0	0	0	0	
		Sum	7	0	23	2	3	35	65.7%

Table 4. ***Cont.***

		Model						
		Gable	**Hip**	**Flat**	**Slant**	**Failure**	**Sum**	**Producer's Accuracy**
Slant	(a)	2	0	0	2	0	4	
	(b)	0	0	0	0	0	0	
	(c)	0	0	0	0	0	0	
	(d)	0	0	0	0	0	0	
	(e)	0	0	0	0	0	0	
	Sum	2	0	0	2	0	4	50.0%
Sum		257	3	50	38	6	354	
User's Accuracy		93.0%	100%	46.0%	5.3%			75.4%

Figure 12. (**a**) Aerial image, (**b**) segmentation results, and (**c**), (**d**) modeling results for another area containing low-rise buildings and high-rise and complex buildings. Wireframe models are as in Figure 10.

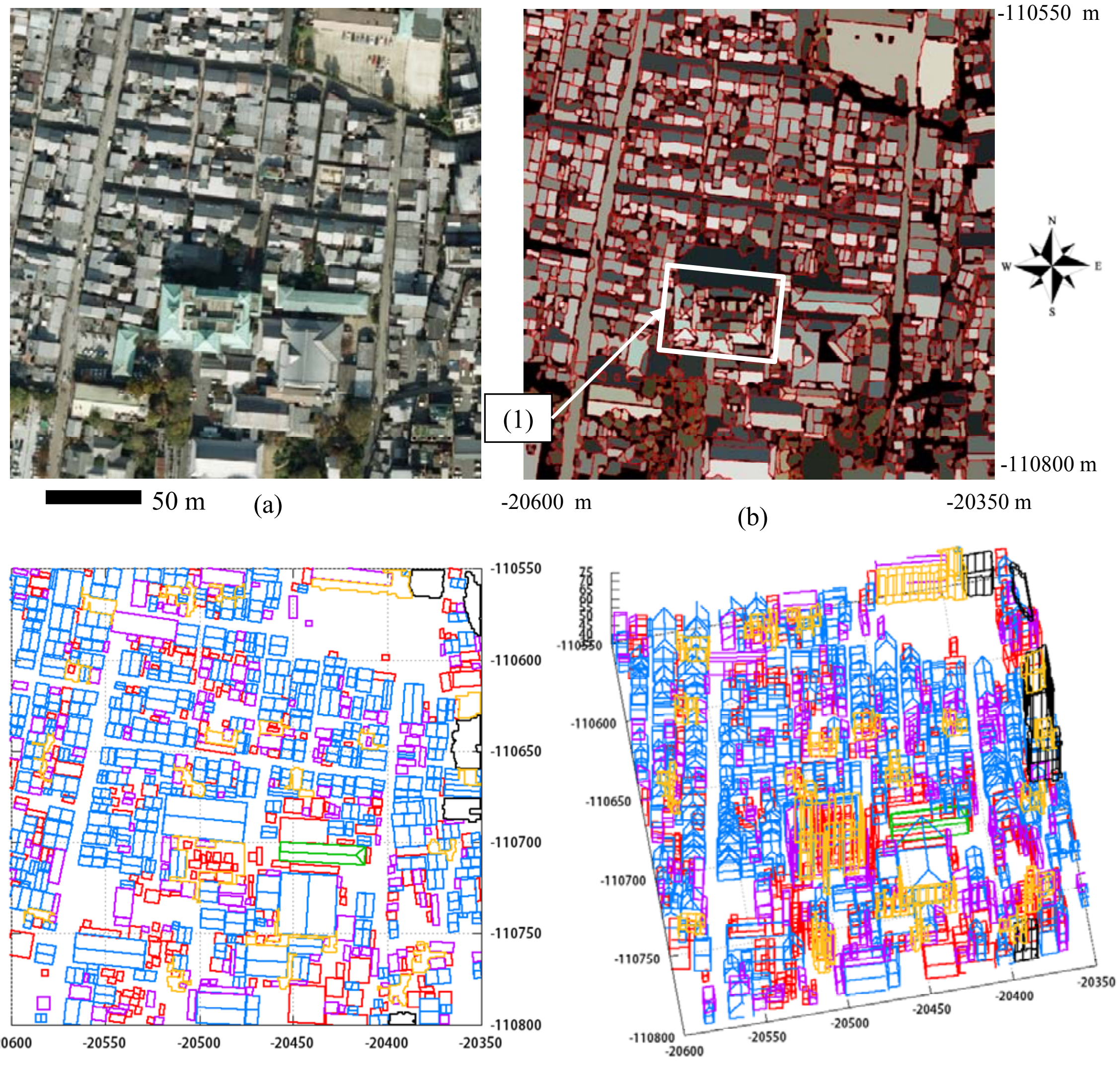
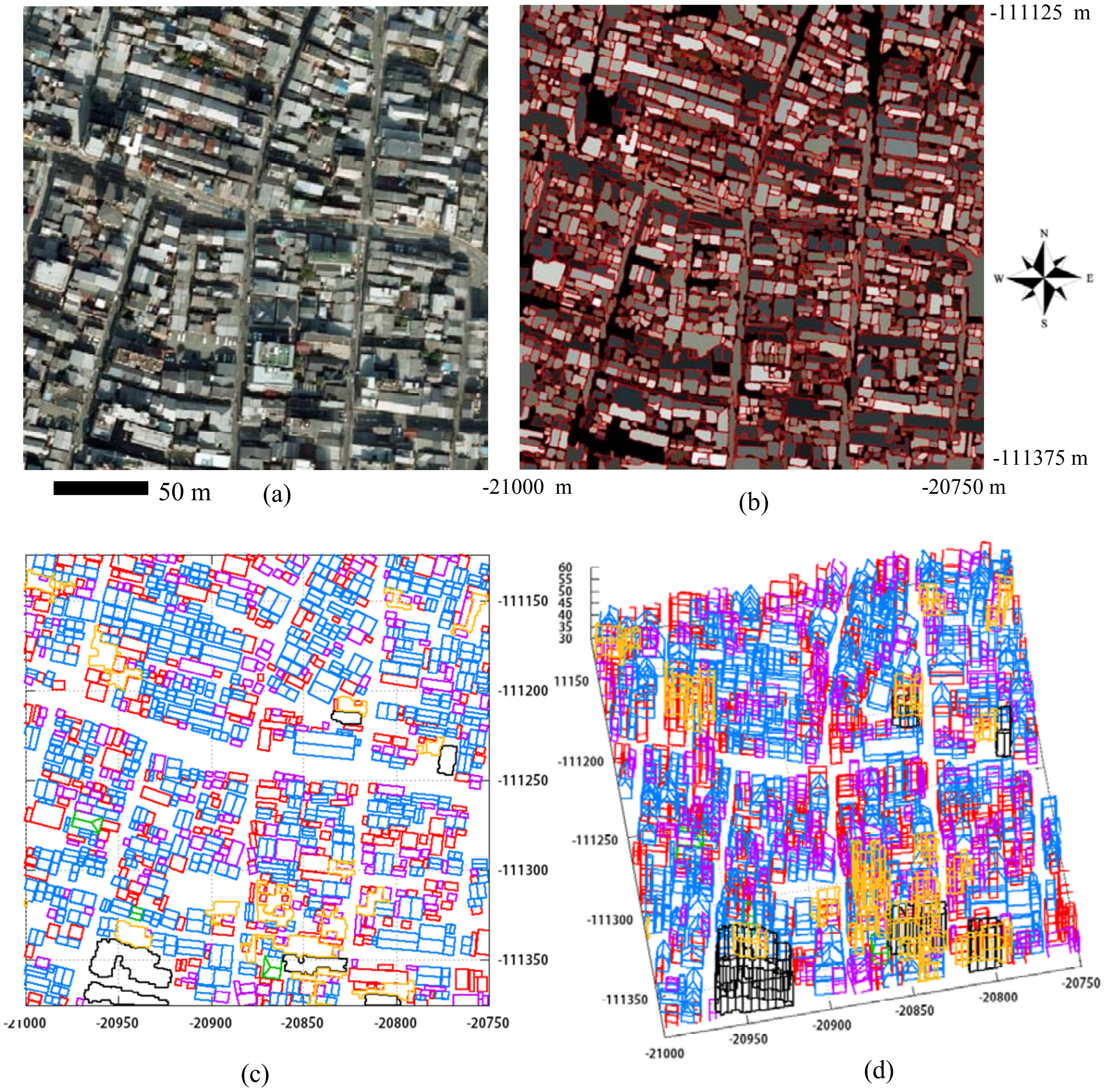


Figure 13. (**a**) Aerial image, (**b**) segmentation results, and (**c**)(**d**) modeling results for another area containing low-rise buildings. Wireframe models are as in Figure 10.



The total error in our 3D models includes both segmentation error and modeling algorithm error. To examine the effect of segmentation error, the relationship between modeling and segmentation performance was investigated. The segmentation results were split into five categories: (a) individual roof planes successfully segmented; (b) all roof planes successfully extracted, but roof planes of the same building merged into one; (c) all roof planes successfully extracted, but roof planes(s) merged with another building's roof(s); (d) not all roof planes extracted; and (e) no roof extracted. The validation results for the study area of Figure 11 are combined into Tables 3 and 4, in which producer's accuracy, user's accuracy, and total accuracy are shown.

After examining the effects of segmentation error, the models were validated in a different manner, by comparing the ground area of the models with areas obtained from manual interpretation of the aerial images (Table 5). Because gable-roof buildings are dominant in this area, gable-roof buildings with actual ground areas of greater than or equal to 50 m^2 and less than or equal to 100 m^2 were randomly selected to perform this validation. Fifty gable-roof buildings were used for this validation.

Table 5. Absolute and relative error of ground area of building model.

Number of samples	Absolute error (m^2)	Relative error (%)
55	10.2	14.0

4. Discussion

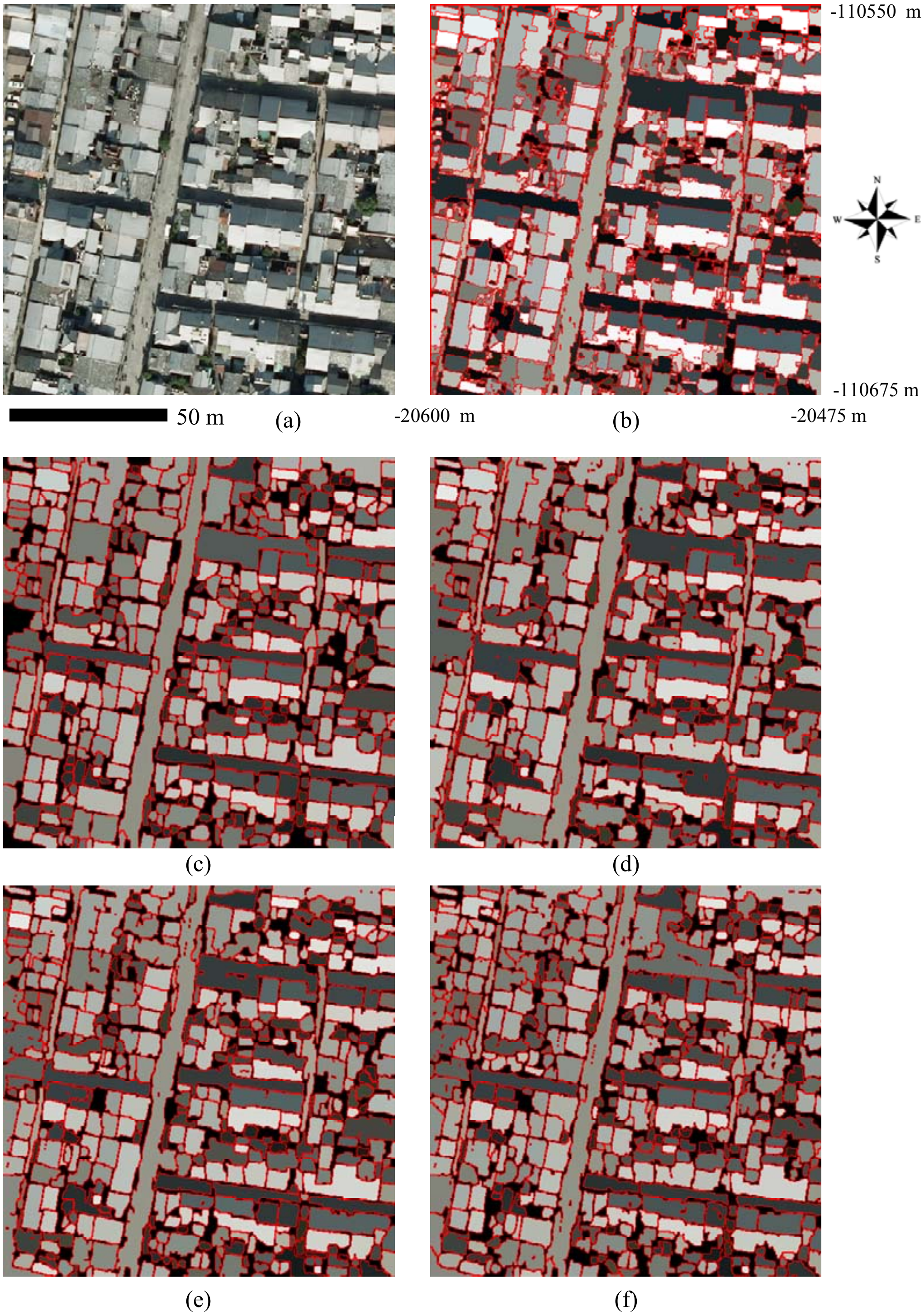
4.1. Effect of Segmentation Quality on Modeling

The proposed algorithm can generate different types of building models, and generated almost the same number of models as actual buildings in the study area. The primary factor in realizing a successful model is appropriate utilization of the region areas provided by the segmentation from aerial images. The primary factor in realizing a successful model is making the best use of the region areas provided by segmentation of aerial images. Figure 14 shows the segmentation results for the upper left area of Figure 12a. The final result (Figure 14c) was generated by selecting regions with higher rectangular index from the regions generated under different discrete widths (Figures 14d–f)). This result shows that the segmentation algorithm [18] outperforms ENVI EX [20] (Figure 14b), and that the segmentation algorithm selects optimal rectangular regions. This indicates that the algorithm can find the optical local smoothing parameter for segmentation. However, the segmentation algorithm has room for improvement, since it failed to exclude road surfaces, and sometimes failed to separate individual roofs. To improve these results, road surfaces were subsequently excluded by counting the effective points that were significantly higher than the ground elevation estimated through filtering.

For the case of failing to separate two-roof regions, later steps in the proposed algorithm generate a gable-roof building model if the regions possess two normals that are linearly symmetric, and generate two pairs of gable roofs when groups of 1 to 2 roofs are found in the segmentation results. By employing these knowledge-based approaches, the proposed algorithm utilizes the geometry of the buildings for modeling, reducing the impact of segmentation errors. Columns (b) and (c) for the gable-roof buildings in Table 4 show that despite segmentation errors, modeling of these buildings was successful. For example, Areas (1) and (2) in Figure 9 show generated models when groups of 1 to 2 roofs were found in the segmentation results, and Areas (3) and (4) show generated models when regions possessed two normals that were linearly symmetric.

Although the segmentation information from aerial images is important, buildings whose regions are not extracted in segmentation can still be modeled in the proposed algorithm. Modeling (4) generates models by using point clouds that are not currently in use in the model, but that indicate that a building most likely exists. Figure 10e shows that the proposed algorithm succeeds in modeling some missing buildings and compensates for segmentation failure. As points used in Modeling (4) may include points reflected from tall vegetation, vegetation removal is executed before modeling. An examination of the areas revealed that in estimating the coefficients for the planar equation, some point clouds reflected from tall trees had small RMSEs. Therefore, the brightness of aerial images could be utilized to avoid generating false models. However, it was difficult to exclude vegetation because of color variety (green, yellow, or red) and crown shadows. Manual removal of vegetation may be painstaking if vegetation areas are not small. In such cases, semi-automatic removal may be reasonable by setting combinations of RGB DN thresholds for green, yellow, and red vegetation, respectively. Techniques for removing vegetation should be improved in the future.

Figure 14. Segmentation results of upper left area of Figure 12a. (**a**) Aerial photograph; (**b**) segmentation result using ENVI EX 4.8; (**c**) segmentation result using the algorithm [18], merged with results with Δd_i = 40, 30, and 20; (**d**) segmentation result with Δd_i = 40; (**e**) segmentation result with Δd_i = 30; and (**f**) segmentation result with Δd_i = 20. Parameter values of "Scale Level" and "Merge Level" in ENVI EX were set to 50 and 80, respectively.



The compensatory factors built into the algorithm also help reduce the effects of the difference in time between the acquisition of LiDAR data and aerial image observations. Whether the image was taken before or after LiDAR observations, certain buildings that exist in the aerial image may not have been captured by LiDAR. Such buildings are removed from the modeling process by our LiDAR filtering (because at the time of LiDAR acquisition, they are in fact ground points). In contrast, if buildings observed by LiDAR are not segmented due to segmentation failure, or are not available in the aerial image due to demolition or construction, these buildings can still be modeled by performing Modeling (4). Therefore, effects due to observation delays can be reduced, and the proposed algorithm can generate models for all buildings available when the LiDAR data are acquired.

A number of point clouds remained even after Step (8) had generated models. Yellow areas in Figure 10d shows that these point clouds correspond to building roof sections. We explored a provisional technique that merged such point clouds with the generated models. However, classification of points into the models was unstable. Therefore, at this moment, a semi-automatic approach to correcting models is reasonable, and a fully automatic approach will be further examined in the future.

4.2. Accuracy Assessment

Table 3 shows that the producer's accuracies for gable-roof, flat-roof, and slant-roof buildings were all within acceptable limits for the models required for a fire spread simulation, whereas the hip-roof producer's accuracy was not acceptable. Table 4 records the conditional probability of generating a building model, given that the building areas are correctly segmented. Flat-roof models generated in Modeling (1) were excluded from Table 4 because they were modeled without using the segmented image. Columns (b) and (c) for the gable-roof show that modeling of gable-roof buildings was successful even when not supported by segmentation results. However, column (a) of the hip-roof results shows that modeling of hip-roof buildings was difficult even with completely successful segmentation. In the unsuccessful cases, the side roof planes of the buildings had a low number of LiDAR points and were not of a single normal type. As a result, the side roof was not regarded as a part of the hip-roof building. Especially in dense urban areas, small roofs may not contain enough data points to calculate their normal, owing to occlusion. The results also show that smaller hip-roof buildings may not be correctly modeled because of the incident angle of the LiDAR; however, this problem can be overcome if higher density airborne LiDAR data is made available for the modeling.

Table 5 shows that the proposed algorithm underestimated the total ground area of the models by 14.0%. When segmenting regions from aerial images in dense urban areas, accurate delineation of building boundaries is a nontrivial problem. Our method corrected the boundaries to eliminate overlapping buildings, as shown in Figure 7. However, this strategy for dealing with overlap generally reduced the area of the overlapping buildings. As a result, the ground areas of models are expected to be underestimated for dense urban areas.

4.3. Variety of Generated Models

The proposed algorithm generates four types of building models: hip-roof, gable-roof, flat-roof, and slant-roof; each has a rectangular rooftop. Some users may not be satisfied with these simple models, especially because more comprehensive model sets have been generated by other research groups [5–7]. However, as mentioned in Section 1, the aim of the present research is to generate models of dense urban areas, for which the algorithms for automatic modeling have been inadequate. The proposed approach overcomes the difficulty of modeling dense urban areas by referring to the regions provided by segmentation from aerial images.

The proposed algorithm modeled complex buildings as one of the four models, a combination of the four models, or a model with some missing parts. For example, L-shaped gable-roof buildings (Area (1) in Figure 11a) were modeled by gable-roof and flat-roof building models in Figure 11c,d. In the area containing low-rise buildings and high-rise and complex buildings, the complex building (Area (1) in Figure 12b) was modeled as a large gable-roof model, and other complex buildings in the area were properly modeled as gable-roof models.

For Kyoto, simple models, rather than detailed models, have been requested by researchers in the field of disaster assessment for modeling the propagation of fire caused by an earthquake. In terms of feasibility, it remains difficult to automatically generate detailed models of dense urban areas, even by combining high-density LiDAR data and aerial images. As a result, the author generated simple models rather than complex models. If users wish to generate detailed models or models with non-rectangular rooftops, it is recommended to determine whether the boundary of the building is nearly rectangular by examining the shape of the segmented region. If it is not nearly rectangular, another model with a non-rectangular rooftop should be applied.

4.4. Determination of Azimuthal Direction of Ridges in Dense Urban Areas

In developing the algorithm, we found that the roof and the ridge directions of buildings aligned in a row tended to be inconsistent when the directions were estimated individually for each building. This misalignment was due in part to the low density of the airborne LiDAR data and in part to the inaccurate separation of points from different roofs or different buildings, as a consequence of incomplete segmentation. To reduce the effects of this inconsistency, a method was implemented, where filtered point clouds were separated into small groups, and the principal azimuthal direction of all the points in a group was used to determine the roof and the ridge directions, and also the boundaries of the buildings. As a result, accurate models of rows of buildings in dense urban areas were generated even when using the low-density airborne LiDAR data.

The restriction of azimuthal roof alignment is useful to apply the proposed algorithm to dense urban areas, where alignments are generally consistent. Note that this approach might not work for a building whose roof's azimuthal direction is completely contradictory with those of other buildings. However, as shown in Figures 11 and 12, such conflicting buildings were not observed in any of the study areas. If a building with a different azimuthal direction exists a few meters away from other buildings, the point clouds associated with that building can be easily separated from the other point clouds, and thus the roof and the ridge directions are accurately estimated. In this regard, in the case of non-dense urban areas where buildings are not close to each other, the algorithm still works because the roof and ridge directions are determined within point clouds classified using Parameter (e), horizontal and vertical distance for segmentation of point clouds. The argument here is that determining a building's azimuthal direction is highly dependent on point clouds being separated building by building, as mentioned in Section 1. Because this separation is difficult for dense urban areas, the approach taken in the proposed algorithm is reasonable.

4.5. Parameter Selection

The parameter-setting process and the sensitivity of the results to the parameter selections are important in terms of transferability of the proposed algorithm. The factors affecting the parameters can be classified as follows: (1) artificial geographical features (e.g. average building area/size, and street width); (2) LiDAR point density; (3) both artificial geographical features and LiDAR point density; and (4) others. Factor (1) is dependent on the study area, and impacts the following parameters: (e) horizontal and vertical distance for segmentation of point clouds, (f) minimum height for discrimination of roof points, (k) minimum ratio between maximum length of the smaller region and that of the larger region, (m) horizontal and vertical distance for segmentation of point clouds, (o) minimum length of rectangle, and (p) minimum area. Factor (2) impacts parameter (a) window size. Parameters affected by Factor (3) are as follows: (d) minimum vertical component of normal for flat roof, (g) minimum probability for single normal class, (h) minimum probability for two normal classes, and (n) minimum number of points. Parameters (i) minimum valid length, (j) maximum invalid length, and (l) ratio *r* for boundary search can be determined by the quality of the segmentation result of aerial image, and its spatial resolution.

In case of the study area, it took 20 to 30 iterations to check the validity of the selected parameters. In particular, the values of Parameters (g) minimum probability for single normal class and (h) minimum probability for normal classes were found to be very sensitive. As it may be difficult to automatically optimize such parameters, empirical optimization may be necessary. If the proposed algorithm is to be applied elsewhere, it will need to use different parameter values. In particular, application to non-dense urban areas may need to tune parameters included in Factors (1) and (3).

4.6. Computation Time

It took approximately 10 min of elapsed time to model one study area (250 m × 250 m); 5 min 30 s for filtering (approximately 1 km × 1 km), 4 min for segmentation using an aerial image (250 m × 250 m), and 1 min 20 s for modeling using filtered LiDAR data and a segmented image (250 m × 250 m). Most of the computation time is spent segmenting the aerial image. Based on this result, it should take approximately 26 min to model a 1 km × 1 km area. The modeling results may require manual corrections. However, even when including the time for manual corrections, it should be possible to model a 1 km × 1 km area within an hour. This should be an acceptable amount of effort for researchers simulating fire spreading.

As described in Section 3.2, our 250 m × 250 m study region was divided into 125 m × 125 m subregions to reduce computation time. If a large area is being modeled (e.g., 1 km × 1 km) such division is necessary. The algorithm should be improved in the future to properly account for buildings that span boundaries between subregions; otherwise, some buildings may be modeled as smaller than the actual area. In addition, the segmentation algorithm should be improved to further reduce computation time and increase model accuracy.

5. Conclusions

In this paper, a knowledge-based algorithm is proposed that automatically generates 3D building models for dense urban areas by using the results of building segmentation from an aerial image. The algorithm is totally automatic except for setting parameters. Because it uses an improved segmentation algorithm that functions even in dense urban areas, more accurate individual building boundaries can be obtained. The proposed algorithm also uses information about real-world spatial relationships to compensate for weaknesses in segmentation and sparseness of LiDAR data, and is robust to cases where the segmentation data are inadequate (points exist that do not correspond to segmented regions). In addition, the algorithm is a multi-strategy approach, rather than a monolithic algorithm, which means that additional types of buildings could be handled by adding building type–specific strategies. The proposed algorithm was applied to two study areas in Higashiyama ward, Kyoto, Japan. Owing to the information about the building regions provided by segmentation, the modeling of dense urban areas was successful. Although, on occasion, two roofs were not successfully segmented, the proposed algorithm attempts to generate accurate models by referring to the normals in the region. Moreover, the proposed algorithm uses point clouds remaining after modeling by segmentation, which might reveal the existence of a building that was not present in the aerial image. This technique thus compensates for segmentation failure and generates models from LiDAR observations when there is a time gap between acquisition of LiDAR data and aerial images.

The modeling results for the two study areas show that our method generates relatively simple models when compared with other techniques in the literature, but these models can help users to generate detailed models manually. Performance validation results showed the completeness was 76.0%, and the correctness was 77.3%. The proposed algorithm underestimated the total ground area of the models by 10.2 m^2 absolute error and 14.0% relative error. The validation results also showed that the proposed algorithm can effectively generate models, especially gable-roof models, even when information from the segmented image was insufficient. High-density LiDAR data can also be used by the proposed algorithm, but the contribution of this extra data to improvement of modeling accuracy, and to more detailed modeling, might be limited for the case of application to dense urban areas. In the future, the underestimation of the ground area of models can be overcome by improving the accuracy of aerial image segmentation.

Acknowledgments

This research was supported by a Grant-in-Aid for Scientific Research (KAKENHI) for Young Scientists (B) (22760393), and by a grant from Japan Construction Information Center. The author expresses gratitude to Wesco Co. Ltd., Okayama, Japan for providing aerial photographs used in experiments.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Himoto, K.; Tanaka, T. Development and validation of a physics-based urban fire spread model. *Fire Saf. J.* **2008**, *43*, 477–494.
2. Nishino, T.; Tanaka, T.; Hokugo A. An evaluation method for the urban post-earthquake fire risk considering multiple scenarios of fire spread and evacuation. *Fire Saf. J.* **2012**, *54*, 167–180.
3. Zhang, K.; Yan, J.; Chen, S.C. Automatic construction of building footprints from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 2523–2533.
4. Zhou, Q.-Y.; Neumann, U. A Streaming Framework for Seamless Building Reconstruction from Large-scale Aerial LiDAR Data. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009, Miami, FL, USA, 20–25 June 2009; 2759–2766.
5. Forlani, G.; Nardinocchi, C.; Scaioni, M.; Zingaretti, P. Complete classification of raw LIDAR data and 3D reconstruction of buildings. *Pattern Anal. Appl.* **2006**, *8*, 357–374.
6. Sampath, A.; Shan, J. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 1554–1567.
7. Kim, K.; Shan, J. Building roof modeling from airborne laser scanning data based on level set approach. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 484–497.
8. Elberink, S.O. Target graph matching for building reconstruction. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2009**, *38*, Part 3/W8, 49–54.
9. Huber, M.; Schickler, W.; Hinz, S.; Baumgartner, A. Fusion of LIDAR Data and Aerial Imagery for Automatic Reconstruction of Building Surfaces. In Proceedings of 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas, 2003, Berlin, Germany, 22–23 May 2003; pp. 82–86.
10. Sohn, G.; Dowman, I. Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction. *ISPRS J. Photogramm. Remote Sens.* **2007**, *62*, 43–63.
11. Rottensteiner, F.; Trinder, J.; Clode, S.; Kubik, K. Building detection by fusion of airborne laser scanner data and multi-spectral images: Performance evaluation and sensitivity analysis. *ISPRS J. Photogramm. Remote Sens.* **2007**, *62*, 135–149.
12. Awrangjeb, M.; Ravanbakhsh, M.; Fraser, C.S. Automatic detection of residential buildings using LIDAR data and multispectral imagery. *ISPRS J. Photogramm. Remote Sens.* **2010**, *65*, 457–467.
13. Awrangjeb, M.; Zhang, C.; Fraser, C.S. Automatic extraction of building roofs using LIDAR data and multispectral imagery. *ISPRS J. Photogramm. Remote Sens.* **2013**, *83*, 1–18.
14. Caceres, J.J.; Slatton, K.C. Improved Classification of Building Infrastructure from Airborne Lidar Data using Spin Images and Fusion with Ground-based Lidar. In Proceedings of 2007 Urban Remote Sensing Joint Event, Paris, France, 11–13 April 2007; pp. 1–7.
15. Lecumberry, F.; Pardo, Á.; Sapiro, G. Simultaneous object classification and segmentation with high-order multiple shape models. *IEEE Trans. Image Proc.* **2010**, *19*, 625–635.
16. Tzotsos, A.; Karantzalos, K.; Argialas, D. Object-based image analysis through nonlinear scale-space filtering. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 2–11.
17. Li, P.; Guo, J.; Song, B.; Xiao, X. A multilevel hierarchical image segmentation method for urban impervious surface mapping using very high resolution imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2011**, *4*, 103–116.

18. Susaki, J. Segmentation of shadowed buildings in dense urban areas from aerial images. *Remote Sens.* **2012**, *4*, 911–933.
19. Susaki, J. Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) Generation. *Remote Sens*. **2012**, *4*, 1804–1819.
20. ENVI EX. *ENVI EX User's Guide*; 2009. Available online: http://www.exelisvis.com/portals/0/pdfs/enviex/ENVI_EX_User_Guide.pdf (accessed on 2 June 2013).