



## INDONESIAN JOURNAL ON GEOSCIENCE

Geological Agency  
Ministry of Energy and Mineral Resources

Journal homepage: <http://ijog.geologi.esdm.go.id>  
ISSN 2355-9314, e-ISSN 2355-9306



### A Simple But Effective Approach of Building Footprint Extraction in Topographic Mapping Acceleration

DANANG BUDI SUSETYO, ALDINO RIZALDY, MOCHAMAD IRWAN HARIYONO, NUGROHO PURWONO,  
FAHRUL HIDAYAT, RIZKA WINDIASTUTI, TIA RIZKA NUZULA RACHMA, and PRAYUDHA HARTANTO

Geospatial Information Agency of Indonesia (BIG)

Corresponding author: [danang.budi@big.go.id](mailto:danang.budi@big.go.id)

Manuscript received: December, 9, 2019; revised: January, 17, 2020;  
approved: October, 29, 2020; available online: August, 10, 2021

**Abstract** - Topographic mapping using stereo plotting is not effective, because it takes much time and labour-intensive. Thus, this research was conducted to find the effective way to extract building footprint for mapping acceleration from LiDAR data. Building extraction method in this process comprises four steps: ground/non-ground filtering, building classification, segmentation, and building extraction. Classification of ground and non-ground classes was performed using Adaptive-TIN Surface algorithm. Non-ground points from filtering process were classified as building with the algorithm based on multiscale local dimensionality to separate points at the maximum separability plane. Segmentation using segment growing was used to separate each building, so boundary detection could be conducted for each segment to create boundary of each building. Lastly, building extraction was conducted through three steps: boundary point detection, building delineation, and building regularization. With ten samples and step 0.5, classification resulted in quality and miss factor of 0.597 and 0.524, respectively. The quality was improved by segmentation process to 0.604, while miss factor was getting worse to 0.561. Meanwhile, on the average shape index value from extracted building had 0.02 difference, and the number of errors was 30% for the line segment comparison. Regarding positional accuracy using centroid accuracy assessment, this method could produce RMSE of 1.169 m.

**Keywords:** building footprint, LiDAR, classification, segmentation

© IJOG - 2021

#### How to cite this article:

Susetyo, D.B., Rizaldy, A., Hariyono, M.I., Purwono, N., Hidayat, F., Windiastuti, R., Rachma, T.R.N., and Hartanto, P., 2021. A Simple But Effective Approach of Building Footprint Extraction in Topographic Mapping Acceleration. *Indonesian Journal on Geoscience*, 8 (3), p.329-343. DOI: [10.17014/ijog.8.3.329-343](https://doi.org/10.17014/ijog.8.3.329-343)

#### INTRODUCTION

The availability of high-quality geospatial data, including building footprint as part of digital topographic map, is necessary for bigger purposes, including 3D city models (Sugihara *et al.*, 2012; Park and Guldman, 2019) and spatial planning. Building footprint in topographic maps depicts the boundary of building roof with specified level of detail. Information about building footprint is important in spatial planning to

understand the location of a settlement area or identification of building functions on a higher scale.

Traditionally, topographic data production uses stereo plotting process utilizing the photogrammetry technique. However, manual stereo plotting takes much time and labour-intensive, so it is not effective to be applied for large scale topographic mapping in the whole area of Indonesia. Due to the high needs for large scale topographic maps, acceleration of large-scale mapping must

be conducted. Therefore, innovation in a large scale mapping method should be discovered.

Automation of map feature extraction is one of the solutions for mapping acceleration. The automation will reduce the amount of time and human resources, so that it will increase the speed of topographic map production. One of the map features that have been evaluated to be extracted in this research is building. It is not easy to develop the algorithm for automatic building extraction, because most buildings in Indonesia are very dense and close each other, and many of them are in irregular shapes, making it a more challenging research.

Research of automatic building extraction is not of something new, many methods have been developed effectively and accurately. A recent technology is based on deep learning algorithms, for example using neural networks and Markov Random Fields (MRF) (Davydova *et al.*, 2016), end-to-end trainable gated residual refinement network (GRRNet) (Huang *et al.*, 2019), Fully Convolutional Network (Antasari, 2019), Convolutional Neural Network (Yang *et al.*, 2018), Deep Convolutional Network (Setiaji and Harintaka, 2019), or Deep Fully Convolutional Networks (Persello and Stein, 2017). Then, the development of LiDAR creates a new approach for automation of building extraction, for instance using OBIA (Object Based Image Analysis) with the use of class modeling methods (Tomljenovic *et al.*, 2016), data fusion of point and grid-based features using a graph cut algorithm (Du *et al.*, 2017), a novel ordered point-aided Hough Transform (OHT) (Widyaningrum *et al.*, 2019), a modified LEGION segmentation (Liu *et al.*, 2012), automatic process with the emphasis on top-down approaches (Huang *et al.*, 2013), and rule-based segmentation of non-ground LiDAR point clouds (Awrangjeb and Fraser, 2013).

The purpose of this research is to find the best and the most effective approach to extract building footprint automatically for topographic mapping acceleration in Indonesia. It is because the most needed in Indonesia is finding a simple but effective approach to extract topographic map features, since completing topographic map for a

large country such as Indonesia is still challenging. This research is expected to improve business process of topographic mapping in Indonesia to be cheaper, faster, and more effective.

## MATERIALS AND METHODS

### Data and Materials

This research used LiDAR data acquired in 2016 (for building footprint extraction) and topographic map of Indonesia at 1:5,000 scale created in 2017 (as reference dataset). More specifically, the research was conducted at Sayang-sayang Village, located in Cakranegara District in Mataram City, Lombok Island, West Nusa Tenggara Province, Indonesia (Figure 1). This area was chosen because it represented various characteristics of buildings, containing both dense and sparse building groups, so that the analysis could be conducted based on the density of the buildings. Regarding the specification of the data, the density of point clouds was 4 ppm (points per m<sup>2</sup>) or higher, meaning that for each square meter of the area there were at least four points. This specification was suitable for Indonesia topographic mapping at the scale of 1:5,000.

### Methods

#### Ground/Non-ground Filtering

Ground/non-ground filtering was one of the important steps in this research, because it would affect building classification result. Ground and non-ground classification was not only useful for DTM extraction, but also for identifying building, because it could reduce the number of points (Ramiya *et al.*, 2017). In that sense, the next step of point cloud classification into vegetation and building classes was conducted in non-ground points only. Therefore, point cloud separation to ground and non-ground classes determined the quality of building classification.

Classification of ground and non-ground classes was performed using Adaptive-TIN Surfaces algorithm in Terrasolid software. It uses densification of the TIN, where a point will be added

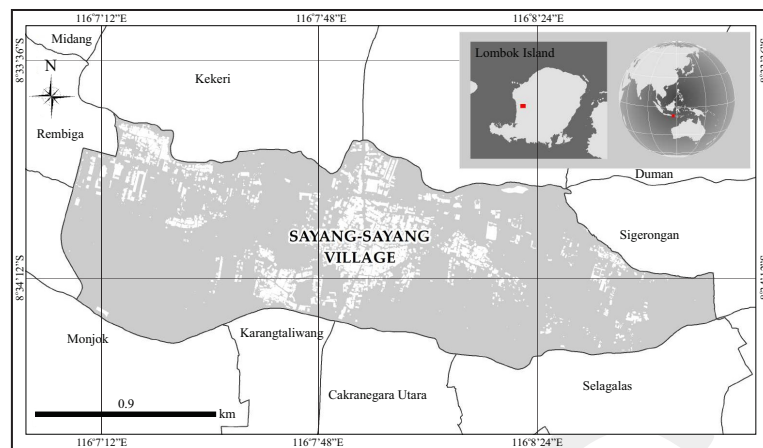


Figure 1. Area of interest, located in Sayang-sayang Village, West Nusa Tenggara Province, Indonesia.

if it meets the calculated threshold parameters, namely distances to the facet planes and angles to the nodes (Figure 2a), and edge areas are cut off if they exceed the threshold (Figure 2b) (Axelsson, 2000). The inputs such as maximal building size, terrain angle, iteration angle, and iteration distance were crucial, in which the values used were 100 m for maximum building size,  $88.0^\circ$  for terrain angle,  $5.00^\circ$  to plane for iteration angle, and 1.40 m for iteration distance. The maximum building size shows the area that has at least one initial ground point, a terrain angle assumes the steepest angle between points in generated TIN, a terrain angle depends on the terrain condition in which a smaller iteration angle is better for a flat terrain and vice versa, and the iteration distance is the maximum distance between point to triangle (Rizaldy and Mayasari, 2016).

#### Building Roof Classification

Building roof classification used the algorithm developed by Brodu and Lague (2012). It was

based on a multiscale local dimensionality to separate points into two classes at the maximum separability plane. The classification considered how the cloud geometrically looked like at ball geometry at a given scale, then the best combination of scales at which dimensionality was measured. The diameter of the ball was defined as a scale, while point of interest was located at the centre, as shown in Figure 3.

To determine whether the cloud appeared locally at a given scale in 1D, 2D, or 3D, the proportion of variance by the eigen values resulting from Principal Component Analysis (PCA) was used, with the Equation (1) defined the proportion of variance (Brodu and Lague, 2012).

$$pi = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} \dots\dots\dots (1)$$

where if  $\lambda_i$ ,  $i = 1 \dots 3$  be the eigen values, ordered by decreasing magnitude:  $\lambda_1 \geq \lambda_2 \geq \lambda_3$ . The number of eigen values was accounted for the total variance

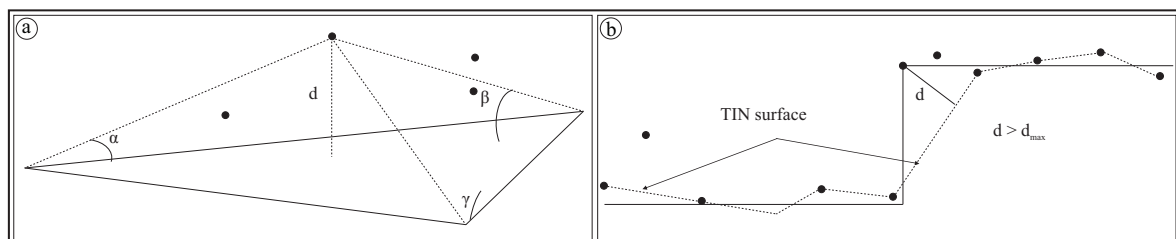


Figure 2 . Parameters of TIN densification in Adaptive TIN Model algorithm (Axelsson, 2000). (a). Distances to the faced planes and angles to the nodes. (b). Edge areas are cut off.

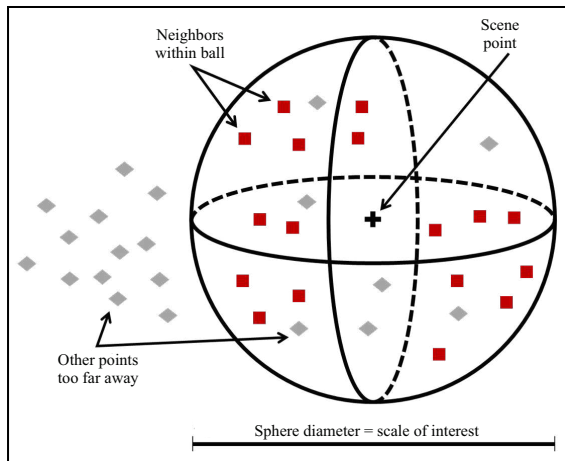


Figure 3. Neighbourhood ball (Brodu and Lague, 2012).

in the neighbourhood ball determined whether the points were distributed in one, two, or three dimensional around the reference scene point.

The algorithm used a linear Support Vector Machines (SVM) created by CANUPO training plugin in Cloud Compare software. A linear classifier proposed one solution in the form of hyperplane that best discriminated and separated two classes (*e.g.* vegetation against building). It used weight vector ( $w$ ) and bias ( $b$ ), in which the hyperplane was defined by equation  $w^T X - b = 0$ .  $w$  and  $b$  were used to project the feature space on the hyperplane, then the distance to the hyperplane  $d_1 = w^T X - b_1$  was calculated for each point. The second distance  $d_2$  was obtained by repeating the process in order to get the second-best direction

orthogonal to the first, then both distance  $d_1$  and  $d_2$  were used as coordinates at 2D plane of maximum separability.

The plane of maximum class separability only kept two main components, so in this research, point clouds were separated into two categories: building and vegetation. To do this classification step, the training area was first created that represented both categories. Then, some parameter numbers were decided to create this training data, including minimum and maximum scale to be the limits of multiscale dimensionality. The unit of the scale referred to the unit of the point cloud. In this research, the chosen number for scales was 2-10, because the building lengths in the researched area were considered between 2 and 10 m, since they were located at a settlement area. Another parameter was step, defined as the distance unit from the minimum to the maximum scales, in order to achieve multiscale features. Small step means more features, otherwise, larger step means less features. In this research, the chosen number for the step was 0.5 m.

Shortly, the whole process for all classification steps is explained in Figure 4. It started from ground classification and non-ground classification, then non-ground points were classified as building and vegetation points. All processes here were binary classification. The building points were then used in the next stage, namely segmentation.

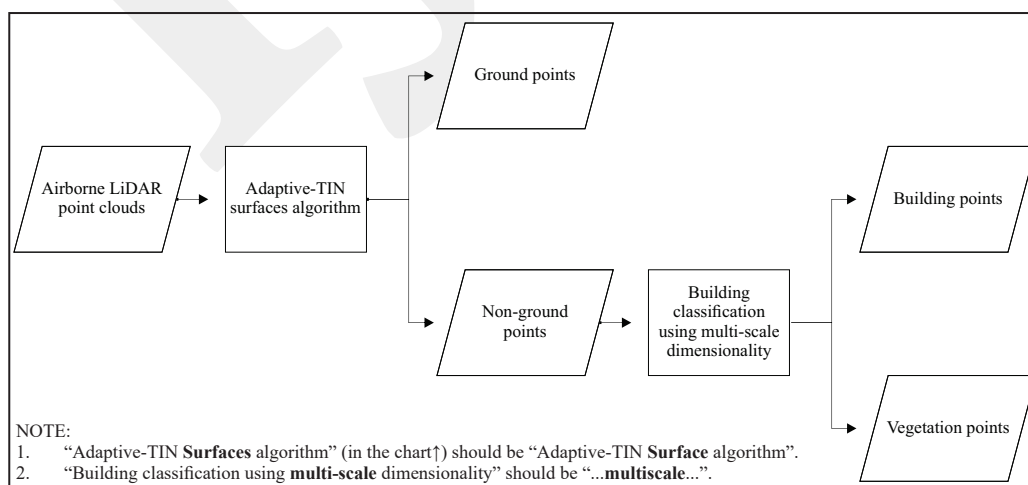


Figure 4. Classification process from unclassified point clouds to classified building points.



### Segmentation

Segmentation in point cloud is used to distinguish group of points based on feature similarity. In order to improve classification result, Vosselman (2013) used segment-based features rather than traditional point-based features for point based classification. The segment-based features led to the more accurate classification result. In this research, segmentation was used to separate each building, so that boundary detection could be conducted per segment to create boundary of each building. The data used in this stage was the previous classification result, namely classified building points.

Segment growing algorithm (Vosselman, 2013) was chosen for the segmentation process. Point Cloud Mapper software was used for the implementation of the growing segmentation. In this algorithm, the test for accepting neighbouring points as extensions of a segment was based on the similarity of feature values and normal vectors scaled by the planarity to distinguish between surfaces with different orientations (Vosselman, 2013). Seed points with similar feature values would be extended and merged with their neighbour points if their values were close to the average feature values for each segment. The combination with the normal vector direction scaled by planarity was used to distinguish between surfaces with different orientations, in which when  $\lambda_1 > \lambda_2 > \lambda_3$  were the eigen values of the co-variance matrix, the planarity could be expressed as  $\frac{(\lambda_2 - \lambda_3)}{\lambda_2}$  (Vosselman, 2013).

Segmentation was also used to remove small segments and unsegmented points. Therefore, it eliminated building errors in the building extraction process. Unsegmented points, coloured as white points (Figure 5) and did not have segment number, were points without sufficient nearby similar points to start a new seed (Vosselman, 2013).

### Building Footprint Extraction

Building extraction process started from boundary detection for each segment to create delineation of coarse building footprint from those boundary points. After segmented points

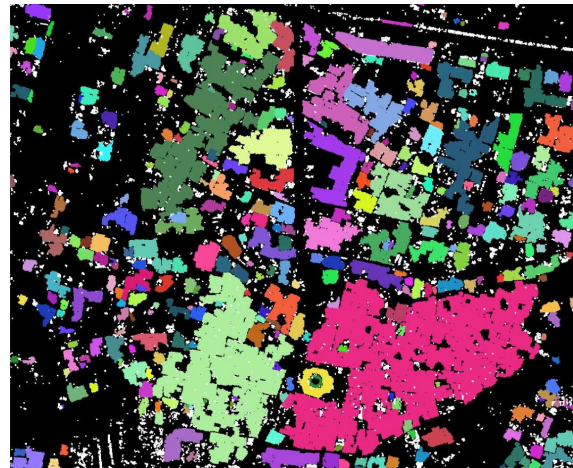


Figure 5. Unsegmented points (coloured as white points) around segmented point clouds.

were obtained using segment growing, then boundary points of each segment were extracted using developed boundary package in MATLAB before delineation of coarse building boundaries was created from those boundary points (Figures 6a, 6b, and 6c). The algorithm of boundary point detection returned points that represented boundary around the building roof points. Unlike the convex hull, the boundary could shrink towards the interior of the hull to envelop the points (<https://www.mathworks.com/help/matlab/ref/boundary.html>). Then, points for each segment were connected with lines using Points to Line tool, so it produced raw building footprints. Lastly, to produce a smooth and sharp building footprint, regularization of building boundary was conducted using Building Regularization tool that applied algorithm developed by Gribov (2015) (Figure 6d). The implementation of the boundary delineation and building footprint regularization was performed in ArcGIS Pro software (<https://www.esri.com/en-us/arcgis/products/arcgis-pro/resources>).

In order to achieve a good geometry of building regularization, reconstruction of coarse building required the support of 90° and 45°. This algorithm was also suitable to arcs, in which the arc passing through considered locations differed from the segment by the necessity to define the radius (Gribov, 2015). Building regularization required the setting of tolerance parameter that was

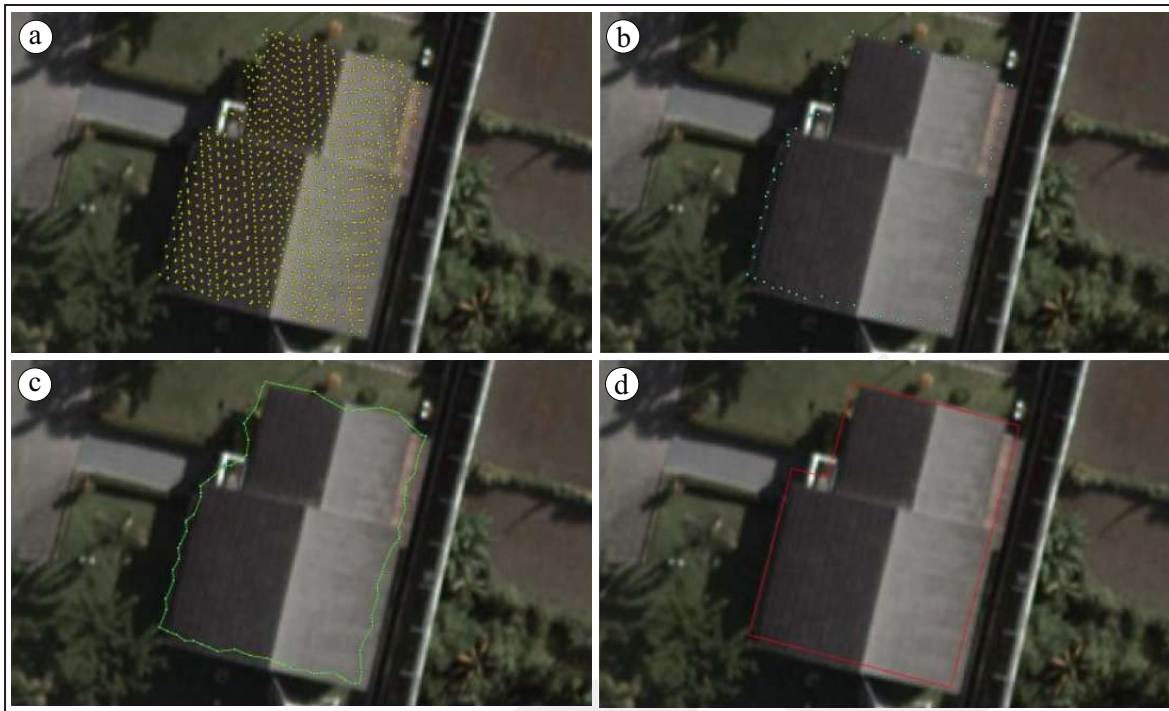


Figure 6. Boundary points of each segment were extracted using developed boundary package in MATLAB. (a) Segmented point cloud from segment growing helped to identify each single building; (b) From segmented point cloud, boundary points were created using developed boundary package from MATLAB; (c) Boundary points were used to delineate coarse building boundary, so coarse boundaries for each building were created; (d) Regularization process to coarse boundary was needed to produce the final building footprint.

set to 1 m. This number was based on experiments using several numbers, considering the difference between the number of segments from extracted and reference segments, so it could be assumed that extracted objects had similar level of detail to reference objects.

## RESULT

Figure 7 illustrates the result of each process, from ground/non-ground classification until building footprint extraction. It can be seen that each process affected the quality of the next process. For instance, due to some very close buildings were segmented as one segment, in building extraction result they became one single building. Thus, to produce a good extraction result, the quality of each process from the beginning must be ensured.

The comparison between extracted and reference data (topographic map) is presented in Figure 8. The green circle shows that the extracted

buildings are similar to reference data, while the yellow one shows that the extracted buildings still need improvement. Some errors happened, for instance, the very close and dense buildings were extracted as one single building (Figures 8a and 8b). Besides the dense area, another challenge was vegetation, where the buildings that were covered by vegetation had problems in their geometric result (Figures 8c and 8d), even not extracted at all (Figure 8e).

## DISCUSSION

### Classification Analysis

The classification result of training data was shown in a plane of maximum separability and separation line (Figure 9). Balanced accuracy (*ba*) and Fisher Discriminant Ratio (*fdr*) were used to quantify the performance of the classifier, where a large *ba* value indicated a good recognition rate, while a large *fdr* value indicated that classes were



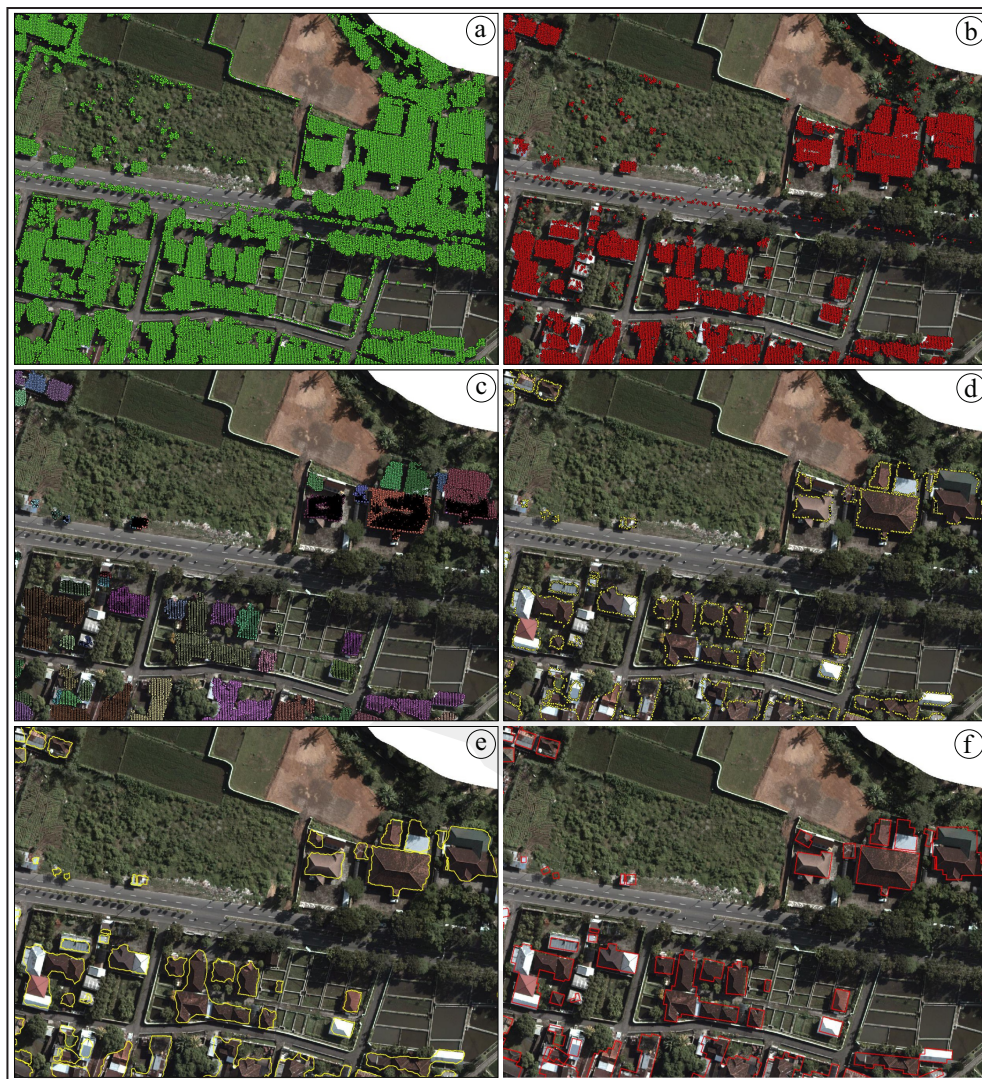


Figure 7. The result of each process: (a) Non-ground points; (b) Building points; (c) Segmented points; (d) Building boundary points; (e) Coarse building line; (f) Final building footprint.

well separated, and obtained using these equations (Brodu and Lague, 2012):

$$ba = \frac{1}{2}(a_v + a_b) \dots\dots\dots (2)$$

where:

- $a_v = \frac{tv}{tv + fb}$
- $a_b = \frac{tb}{tb + fv}$
- $tv, tb, fv, fb$  = the number of points truly(*t*) / falsely(*f*) classified into the vegetation(*v*) / building(*b*) classes.

$$fdr = (\mu_2 - \mu_1)^2 / (v_1 + v_2) \dots\dots\dots (3)$$

where:

$\mu_c$  and  $v_c$  = the mean and variance of the signed distance *d* for each class *c*

To test the effect of the number of samples and step parameter to *ba* and *fdr* results, experiments with various parameters were conducted. Several numbers of samples were used from 10, 15, 20, 25, and 30, and also changed the step parameter by 0.5 and 1, where the result was shown in Table 1. It can be seen that the addition of samples and changing step parameters does not improve the results of *ba* and *fdr* too much.



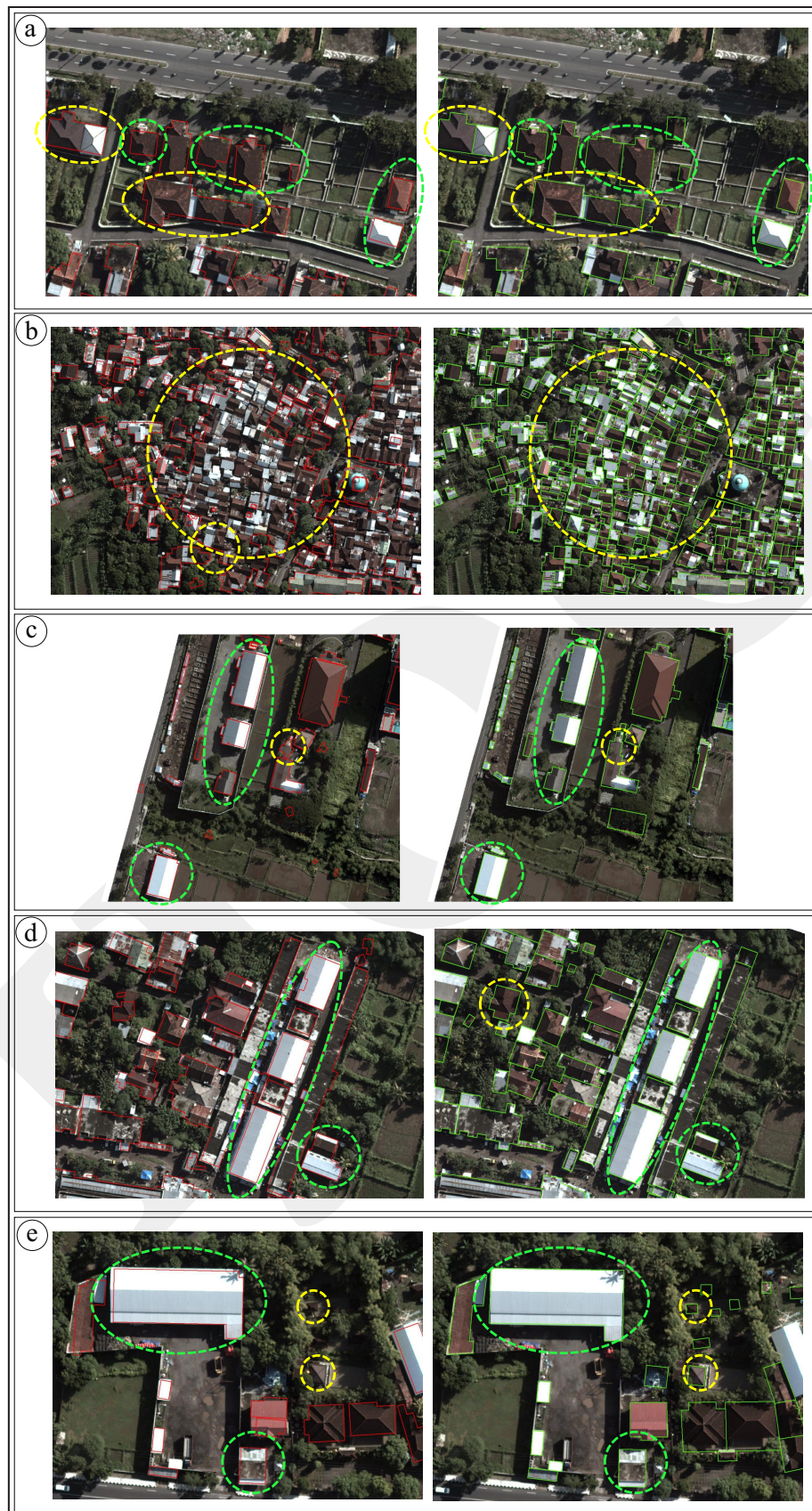


Figure 8. Comparison between extracted data and reference data (topographic map at the scale of 1:5,000). (Explanation for each figure see on text of Result).

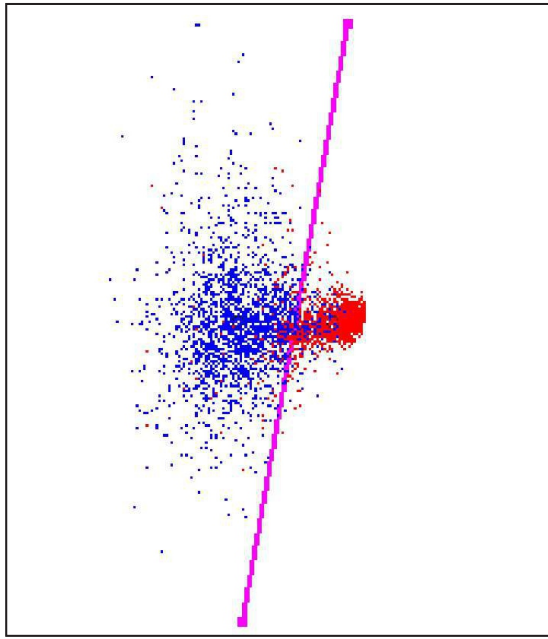


Figure 9. Classifier definition in the plane of maximum separability. Blue: vegetation samples, red: building samples. From statistic calculation, the value of *ba* was 0.916, while the value of *fdr* was 3.811.

Table 1. Results of *ba* and *fdr* Value with Various Numbers of Samples and Step Parameters

Number of Samples	Step	<i>ba</i>	<i>fdr</i>
10	0.5	0.916	3.811
	1	0.912	3.414
15	0.5	0.926	3.651
	1	0.923	3.763
20	0.5	0.918	3.583
	1	0.909	3.213
25	0.5	0.92	3.463
	1	0.909	3.218
30	0.5	0.916	3.344
	1	0.914	3.203

Besides comparing the *ba* and *fdr* values, analysis of classification accuracy was conducted using matched rate analysis, consisting of completeness, correctness, quality, and miss factor, defined by equation (4) - (7) (Zeng *et al.*, 2013). Initially, those equations were used to assess extracted buildings instead of classified building points, but in this research, it was adapted to examine the accuracy of building classification result. Reference points here were non-ground point clouds that were overlapped with building geometry from the topographic map, which were obtained from manual stereo plotting. It was assumed that those points were the correct building points and there

was no geometrical offset between point cloud and the topographic map.

$$Completeness = \frac{TP}{TP + FN} \dots\dots\dots (4)$$

$$Correctness = \frac{TP}{TP + FP} \dots\dots\dots (5)$$

$$Quality = \frac{TP}{TP + FN + FP} \dots\dots\dots (6)$$

$$Miss\ factor = \frac{FN}{TP} \dots\dots\dots (7)$$

where:

- TP (true positive) = the common points of classified points and reference points
- FP (false positive) = the classified points but not reference points
- FN (false negative) = the points that belong to the reference but not the classified points
- Completeness = the percent of classified points being correctly detected with respect to the reference data
- Correctness = the percent of correctly detected part to the classified points
- Quality = metric that combines the completeness and correctness
- Miss factor = the rate of missed building points

The result of matched rate analysis is shown in Table 2. With ten samples and step 0.5, it produced the highest correctness at once the lowest completeness and quality compared to the other parameters. The best result was produced using twenty-five samples and step 1, with quality of 0.666 and miss factor of 0.263. However, since the purpose of this research is to find the simple and effective method for building extraction, the least number of samples was chosen.

### Segmentation Analysis

Segment growing result showed point clouds were segmented to each building, so that segmentation result could be identified as building separation. This process was also used to remove



Table 2. Result of Matched Rate Analysis

Number of Samples	Step	Completeness	Correctness	Quality	Miss Factor
10	0.5	0.656	0.869	0.597	0.524
	1	0.719	0.846	0.635	0.392
15	0.5	0.695	0.849	0.619	0.438
	1	0.740	0.831	0.643	0.352
20	0.5	0.787	0.802	0.659	0.270
	1	0.792	0.800	0.661	0.263
25	0.5	0.790	0.806	0.664	0.266
	1	0.792	0.807	0.666	0.263
30	0.5	0.773	0.822	0.663	0.293
	1	0.769	0.826	0.662	0.300

small segments and unsegmented points to eliminate small and unidentified objects that classified as errors.

Compared to the classification result, segmentation increased the quality of building points based on the assessment method in the previous chapter. The quality was escalated from 0.597 to 0.604, while the correctness was improved from 0.869 to 0.913. Nevertheless, the completeness and miss factor became worse, where the completeness was reduced from 0.656 to 0.641, while miss factor was increased from 0.524 to 0.561. It was because there were more points eliminated in this stage. Therefore, the percentage of missing objects was higher.

### Building Extraction Analysis

Three methods were used to assess the results of geometry of footprint building extraction, which were segment of line comparison, shape index similarity (Veltkamp and Hagedoorn, 2000; Veltkamp, 2011b; Kweon *et al.*, 2019), and edge point-correspondence (Ramirez and Ali, 2003; Kweon *et al.*,

2019). The segment of line comparison was used to evaluate the number of sections of every line resulted from samples against reference dataset. Statistically, both data were calculated to generate distinction of the line segments and counted it into a percentage which indicated the level of error. The matrix of the line segment comparison is shown in Table 3. The shape index similarity was used to enumerate the shape ratio between two objects. This method expressed the ratio of an area to the length of the circumference (Veltkamp, 2011a). Each sample of extracted building was calculated as a polygon, where the shape index was calculated between perimeters in giving an internal area of a polygon. If the shape index was 1, it meant the polygon calculated was circular, while the higher number of 1 indicated the bar-shaped of the polygon. If the polygon was square, it had an index of approximately 1.13. It is possible to express the shape index equation as follows:

$$Di = \frac{pi}{2\sqrt{A\pi}} \dots\dots\dots (8)$$

Table 3. Comparison of Segment Number between Samples of Reference Datasets and Building Extraction Data

A				B				C				D				E			
4				4				12				4				6			
4	4	4	4	4	4	4	4	8	6	4	4	4	4	4	4	8	6	4	4
F				G				H				I				J			
9				4				12				8				6			
6	4	4	4	6	4	4	4	6	4	4	4	10	6	4	4	6	4	4	4
K				L				M				N				O			
4				14				4				9				17			
4	4	4	4	6	4	4	4	4	4	4	4	18	10	6	4	14	10	8	8
P				Q				R				S				T			
8				7				4				10				13			
13	12	6	4	8	6	6	6	4	4	4	4	4	4	4	4	8	6	4	4

where:

$Di$  : index  
 $pi$  : perimeter (m)  
 $A$  : area (m<sup>2</sup>)

Table 3 shows the number of line segment in each data sample, where it was arranged downward respectively in sample ID, the number of segments of reference datasets, and the number of segments of extracting data. Each number of segments of extracting data is shown as four columns as well as the interval of tolerance in building regularization processes, which are 1 m, 1.5 m, 2 m, and 2.5 m consecutively. The comparison of extracting data to reference datasets obtained the maximum number of errors in each sample data. The error was depicted as a percentage which is illustrated in Figure 10a. Based on the graph, the maximum error was found in sample data N, where it had indefectible error regarding to line segment of building extraction. The most suitable interval tolerance value in regularization process was 1 m. For all, the number of errors on the average was 30% or 0.3, meaning the line segment of this building extraction is acceptable.

The assessment of shape similarity resulted in the value of index in every sample. The result of shape similarity index is shown in Figure 10b. Related to extracted data and reference dataset, there was a low difference index mostly. On the average, the whole sample had the difference of 0.02 in the shape index value. Comparing the

whole data, sample S merely generated the highest difference shape index, or quantitatively was 0.3. Thus, the result of building extraction had a nearly similar shape to reference data in particular of building edge features.

The last test was positional accuracy using the calculation of Root Mean Square Error (RMSE). The accuracy was assessed by comparing centroid positions between extracted and reference sample objects. Centroid was used because Euclidean distance between the centres of the mass of an extracted object and the reference object could be used as another metric to measure the positional accuracy other than the corner points of buildings (Zeng *et al.*, 2013). From those buildings, centroid coordinates were calculated on both data as shown in Table 4, RMSE was then obtained from Equation (9).

$$RMSE = \sqrt{\frac{\Delta X^2 + \Delta Y^2}{n}} \dots\dots\dots (9)$$

where:

$n$  = the number of points.

### Time Processing Analysis

These works were conducted using laptop with specification of processor Intel(R) Core (TM) i7-7500U, RAM memory 8.00 GB, and 64-bit Operating system. Thus, the result of this analysis can be a consideration for practicality evaluation using the hardware specification, and the process could be faster using higher comput-

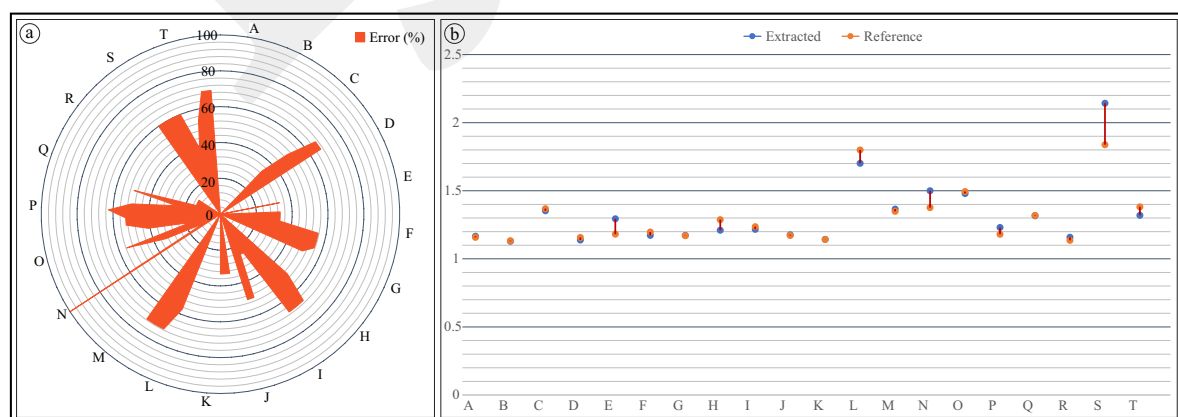


Figure 10. Graphs showing: (a) Maximum error of every extracted data; (b) Shape index similarity difference.

Table 4. Positional Accuracy of Each Building

Building	RBI Centroid		Extracted Centroid		$\Delta X$	$\Delta Y$
	X	Y	X	Y		
A	405948.563	9052575.401	405948.790	9052574.619	0.226	-0.782
B	403751.825	9052608.894	403751.946	9052609.337	0.120	0.443
C	404762.929	9052628.274	404763.587	9052627.533	0.657	-0.740
D	405676.456	9052661.92	405675.447	9052661.517	-1.009	-0.402
E	404172.293	9052784.286	404172.047	9052785.029	-0.245	0.743
F	403496.562	9052842.961	403495.414	9052844.058	-1.148	1.096
G	403047.675	9052971.569	403047.715	9052972.250	0.040	0.680
H	404773.566	9052991.895	404774.698	9052992.894	1.131	0.998
I	404352.525	9053041.472	404353.329	9053041.220	0.804	-0.251
J	403099.219	9053048.893	403098.927	9053049.127	-0.291	0.233
K	403916.275	9053055.857	403916.374	9053055.938	0.099	0.081
L	403215.258	9053062.684	403214.743	9053060.015	-0.514	-2.669
M	403077.960	9053076.599	403077.712	9053076.649	-0.247	0.049
N	404339.570	9053067.305	404340.526	9053068.846	0.956	1.541
O	403728.329	9053078.913	403728.089	9053077.952	-0.239	-0.961
P	403107.326	9053089.027	403106.505	9053089.452	-0.821	0.424
Q	403892.048	9053111.929	403892.200	9053111.525	0.151	-0.403
R	403455.359	9053116.471	403457.022	9053115.892	1.663	-0.579
S	403058.291	9053112.110	403058.095	9053112.389	-0.195	0.278
T	403300.187	9053175.343	403301.247	9053174.690	1.059	-0.652
						RMSE = 1,168

er specification. Time processing analysis was conducted for each step, namely ground/non-ground classification, building classification, segmentation, and building footprint extraction for an area of 2.478 km<sup>2</sup>. Based on the existing topographic map, it has 3,637 buildings. The result of time processing analysis is presented in Table 5.

To assess the effectiveness of this proposed method, manual digitization was conducted in the same area. The result was manual digitization which needs 5 hours and 20 minutes, with

the output specification for manual digitization was similar with the output from automatic extraction (for instance, in a very dense settlement area, delineation was conducted in the boundary of the settlement area). The time processing analysis indicated that the method was almost nine times faster than the manual method, and although this proposed method was simple, it produced good results with high effectiveness. The comparison of manual and automatic extraction for the whole researched area is presented in Figure 11.

Table 5. Results of Speed Processing Analysis

	Process	Times	Information	Software
1.	Ground / non-ground classification	1 min 51.32 sec	1.a. Adaptive-TIN Surfaces algorithm	Terrasolid
2.	Building classification	7 min 20.81 sec	2.a. Training area selection	Cloud Compare
		0 min 24.54 sec	2.b. Generation of training data	Cloud Compare
		7 min 57.62 sec	2.c. Implementation to whole area	Cloud Compare
3.	Segmentation	3 min 33.85 sec	3.a. Processing of segment growing	Point Cloud Mapper
4.	Building footprint extraction	16 min 40.22 sec	4.a. Boundary detection	MATLAB
		0 min 31.95 sec	4.b. Building delineation	ArcGIS Pro
		0 min 43.13 sec	4.c. Building footprint regularization	ArcGIS Pro
	<b>TOTAL</b>	<b>37 min 12.12 sec</b>		

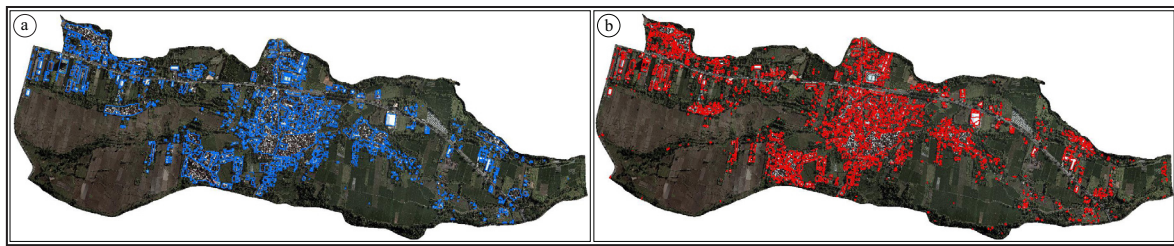


Figure 11. Results of: (a). Visualization of manual digitization and (b). Automatic extraction.

## CONCLUSIONS

This research aims to support mapping acceleration, especially for topographic map production. It focused on automatic building extraction from LiDAR data using the combination of several methods, namely classification, segmentation, and building footprint extraction.

Classification of ground and non-ground classes was performed using Adaptive-TIN Surface algorithm. Building class classification used multiscale local dimensionality to separate points into two classes at the maximum separability plane. Then, segmentation was conducted using segment growing algorithm. Building footprint extraction comprises three steps: boundary point detection, building delineation, and building regularization.

With ten samples and step 0.5, classification resulted in quality and miss factor of 0.597 and 0.524, respectively. The quality was improved by segmentation process to 0.604, while miss factor was getting worse to 0.561. Meanwhile, on the average shape index value from extracted building had the difference of 0.02 and the number of errors was 30% for the line segment comparison. Regarding positional accuracy using centroid accuracy assessment, this method could produce RMSE of 1.169 m. From those statistical analyses, it can be concluded that this method produced acceptable quality for building footprint extraction, especially in sparse settlement areas.

## ACKNOWLEDGMENT

The authors acknowledge Geospatial Information Agency for providing the data used in this

research. The authors also acknowledge ITC – University of Twente, especially Dr. Sander Oude Elberink for giving LiDAR training in Indonesia, and also for providing the software.

## REFERENCES

- Antasari, I.Y., 2019. *Building Detection from Coarse Airborne Laser Scanner Data*. University of Twente.
- Awrangjeb, M. and Fraser, C., 2013. Rule-based segmentation of LiDAR point cloud for automatic extraction of building roof planes. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*. II-3(November), p.1-6. DOI: 10.5194/isprsannals-II-3-W3-1-2013.
- Axelsson, P., 2000. DEM generation from laser scanner data using adaptive TIN models. *International Archives of Photogrammetry and Remote Sensing*, 23 (B4), p.110-117. DOI: 10.1016/j.isprsjprs.2005.10.005.
- Brodu, N. and Lague, D., 2012. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. *ISPRS Journal of Photogrammetry and Remote Sensing*. International Society for Photogrammetry and Remote Sensing, Inc. 68, p.121-134. DOI: 10.1016/j.isprsjprs.2012.01.006.
- Davydova, K., Cui, S., and Reinartz, P., 2016. Building footprint extraction from digital surface models using neural networks. *Image and Signal Processing for Remote Sensing XXII*. p.1-10. DOI: 10.1117/12.2240727.

- Du, S., Zhang, Y., Zou, Z., Xu, S., He, X., and Chen, S., 2017. Automatic building extraction from LiDAR data fusion of point and grid-based features. *ISPRS Journal of Photogrammetry and Remote Sensing. International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS)*. 130, p.294-307. DOI: 10.1016/j.isprsjprs.2017.06.005.
- Gribov, A., 2015. Searching for a compressed polyline with a minimum number of vertices, arXiv.
- Huang, H., Brenner, C., and Sester, M., 2013. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 79, p.29-43. DOI: 10.1016/j.isprsjprs.2013.02.004.
- Huang, J., Zhang, X., Xin, Q., Sun, Y., 2019. Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 151, p.91-105. DOI: 10.1016/j.isprsjprs.2019.02.019.
- Kweon, H., Kim, M., Lee, J-W., 2019. Comparison of horizontal accuracy, shape similarity, and cost of three different road mapping techniques. *Forest*, p.1-13. DOI: 10.3390/f10050452.
- Liu, C., Shi, B., Yang, X., Li, N., 2012. Automatic buildings extraction from LiDAR data in urban area by neural oscillator network of visual cortex. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, p.1-12.
- Park, Y. and Guldmann, J., 2019. 3D city models with building footprints and LIDAR point cloud classification : A machine learning approach. *Computers, Environment and Urban Systems*, 75, p.76-89. DOI: 10.1016/j.compenurbsys.2019.01.004.
- Persello, C. and Stein, A., 2017. Deep Fully Convolutional Networks for the Detection of Informal Settlements in VHR Images. *IEEE Geoscience and Remote Sensing Letters*, 14 (12), p. 2325-2329.
- Ramirez, J.R. and Ali, T., 2003. Progress in Metrics Development to Measure Positional Accuracy of Spatial Data. *Proceedings of the 21<sup>th</sup> International Cartographic Conference (ICC)*. Durban, South Africa.
- Ramiya, A.M., Nidamanuri, R.R., and Krishnan, R., 2017. Segmentation based building detection approach from LiDAR point cloud. *The Egyptian Journal of Remote Sensing and Space Sciences*. National Authority for Remote Sensing and Space Sciences. 20 (1), p.71-77. DOI: 10.1016/j.ejrs.2016.04.001.
- Rizaldy, A. and Mayasari, R., 2016. Acceleration of topographic map production using semi-automatic DTM from DSM radar data, *ISPRS - International Archives of the Photogrammetry. Remote Sensing and Spatial Information Sciences*, XLI-B7, p.47-54. DOI: 10.5194/isprsarchives-XLI-B7-47-2016.
- Setiaji, D. and Harintaka, 2019. Ekstraksi fitur bangunan secara cepat pada foto UAV menggunakan metode deep residual neural network berbasis FCN. *Elipsoida*, 02 (01), p.42-49.
- Sugihara, K., Zhou, X., and Murase, T., 2012. Knowledge-Based System for Automatic 3D. *Proceedings of the 5<sup>th</sup> International Conference on Intelligent Interactive Multimedia Systems and Services*. Springer. p.363-373.
- Tomljenovic, I., Tiede, D., and Blaschke, T., 2016. A building extraction approach for airborne laser scanner data utilizing the object based image analysis paradigm. *International Journal of Applied Earth Observation and Geo-information*, 52, p.137-148. DOI: 10.1016/j.jag.2016.06.007.
- Veltkamp, R.C., 2011a. Shape Matching : Similarity Measures and Algorithms. *Proceedings of International Conference on Shape Modeling and Applications*. Genova, Italy.
- Veltkamp, R.C., 2011b. Shape Matching: Similarity Measures and Algorithms. *Proceedings of International Conference on Shape Modeling and Applications*. Genova, Italy.
- Veltkamp, R.C. and Hagedoorn, M., 2000. Shape Similarity Measures, Properties and Constructions. In: Laurini, R. (ed.), *Advances in Vi-*



- sual Information Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg. p.467-476.
- Vosselman, G., 2013. Point Cloud Segmentation for Urban Scene Classification. In: *ISPRS2013-SSG*. Antalya, Turkey. DOI: 10.5194/isprsarchives-XL-7-W2-257-2013.
- Widyaningrum, E., Gorte, B., and Lindenbergh, R., 2019. Automatic building outline extraction from ALS point clouds by ordered points aided hough transform. *Remote Sensing*. DOI: 10.3390/rs11141727.
- Yang, H.L., Yuan, J., Lunga, D., Laverdiere, M., Rose, A., Bhaduri, B., 2018. Building extraction at scale using convolutional neural network: mapping of the United States. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. IEEE. PP, p.1-15. DOI: 10.1109/JSTARS.2018.2835377.
- Zeng, C., Wang, J., and Lehrbass, B., 2013. An evaluation system for building footprint extraction from remotely sensed data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 6 (3), p.1640-1652.