International Journal of
Geo-Information

MDPI

Article

Machine Learning Classification of Buildings for Map Generalization

Jaeeun Lee [1], Hanme Jang [1], Jonghyeon Yang [1] and Kiyun Yu [1,2,*]

[1] Department of Civil and Environmental Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Korea; perfectsolo@snu.ac.kr (J.L.); janghanie1@snu.ac.kr (H.J.); yangjonghyeon@snu.ac.kr (J.Y.) ; kiyun@snu.ac.kr (K.Y.)

[2] Institute of Construction and Environmental Engineering, Seoul National University, 1, Gwanak-ro, Gwanak-gu, Seoul 08826, Korea

* Correspondence: kiyun@snu.ac.kr; Tel.: +82-02-880-1355

Received: 3 August 2017; Accepted: 11 October 2017; Published: 18 October 2017

Abstract: A critical problem in mapping data is the frequent updating of large data sets. To solve this problem, the updating of small-scale data based on large-scale data is very effective. Various map generalization techniques, such as simplification, displacement, typification, elimination, and aggregation, must therefore be applied. In this study, we focused on the elimination and aggregation of the building layer, for which each building in a large scale was classified as “0-eliminated,” “1-retained,” or “2-aggregated.” Machine-learning classification algorithms were then used for classifying the buildings. The data of 1:1000 scale and 1:25,000 scale digital maps obtained from the National Geographic Information Institute were used. We applied to these data various machine-learning classification algorithms, including naive Bayes (NB), decision tree (DT), *k*-nearest neighbor (*k*-NN), and support vector machine (SVM). The overall accuracies of each algorithm were satisfactory: DT, 88.96%; *k*-NN, 88.27%; SVM, 87.57%; and NB, 79.50%. Although elimination is a direct part of the proposed process, generalization operations, such as simplification and aggregation of polygons, must still be performed for buildings classified as retained and aggregated. Thus, these algorithms can be used for building classification and can serve as preparatory steps for building generalization.

Keywords: aggregation; building generalization; classification; elimination; map generalization; machine learning

1. Introduction

Background map data comprise the most essential features of geographical information systems (GISs) and location-based services (LBSs). Many map providers, including national mapping agencies, produce various map data at different scales and themes. A challenging problem for these map providers is the updating and generating of the most current data. Map data should be frequently updated because of changes on the Earth’s surface that may result from either natural phenomena or human activities. However, the current updating processes are labor intensive and time consuming. The most efficient means of updating maps is to update small-scale maps based on current large-scale maps. The process of transforming large-scale maps into small-scale maps is called map generalization. Map generalization is intended to improve the readability of maps and maintain essential information in this process. Typical map generalization operators include simplification, displacement, elimination, and aggregation. Elimination is needed to remove small buildings, such as sheds or isolated buildings. Displacement is needed to separate buildings that would be too close to each other in the desired map scale, or to move buildings further from roads. Aggregation groups buildings into larger units of built-up blocks if the buildings are not to be separately shown.

1.1. Generalization

Since the 1960s, many studies have been conducted on automated map generalization in the GIS/cartography field. The above operators were developed based on these studies, which determined how map features should be represented at a small scale. In addition, the application of the operators determines which features should be changed and how they should be represented. In most map generalization studies, researchers select appropriate operators for map features, such as roads and buildings, and then apply them with thresholds. The results of map generalization are evaluated with quantitative and qualitative measures.

Several studies have been conducted that focused on building generalization. The use of a clustering method for building aggregation was attempted by Regnauld [1] and Anders and Sester [2]. These respective studies attempted to cluster buildings using a hierarchical clustering algorithm and achieved satisfactory results. Meanwhile, Ware and Jones introduced simulated annealing to handle the displacement of buildings while preserving their alignment [3]. Automated building generalization was proposed by Li et al. [4] and Wang and Doihara [5] with different results. Li et al. suggested automated building generalization based on urban morphology and gestalt theory for the building grouping process, and Wang and Doihara strived to sequentially generalize roads and buildings. In the latter study, building generalization mainly focused on aggregation.

Another study for aggregation was conducted by Damen et al. [6], who adopted morphological operator closures and openings to achieve high quality aggregation of buildings. Park attempted to formulate the generalization criteria required to construct a 1:5000 scale digital map by using a 1:1000 scale digital map [7]. Moreover, a recent study by Vetter et al. [8] strived to automate generalization of individual polygon house features of the Swiss Topographic Landscape Model (TLM). To this end, they used ArcGIS ModelBuilder and combined the operators provided by ArcGIS. These studies or methods showed high performances in their test datasets; however, it remains undetermined if they are effective for other datasets, such as larger ones.

1.2. Previous Studies on Machine Learning

In the present study, a machine learning technique was used for more effective map generalization in a large area. Machine learning is a computer science subfield. It is defined by Samuel [9] as "computers having the ability to learn without being explicitly programmed." The computer formulates generalized rules from large amounts of data through machine learning. The core of the machine learning technique is the formulation of inferences similar to those made by humans based on derived rules. From among the various machine learning methods, we employed in this study a supervised learning approach. Supervised learning is a method by which a computer learns data for each input x and labels it as y. This technique is divided into classification and regression according to the problem type. Typical classification problems include face recognition, speech recognition, and image classification. An example of regression, on the other hand, includes the estimation of the real estate value according to the given area. Several studies have likewise been performed in the GIS field utilizing machine learning techniques.

Balboa and López [10] used an artificial neural network (ANN) for road line classification. They extracted features that can categorize lines and sections through principal component analysis. In addition, they used five road categories as qualitative information. The output class was divided into five classes: very smooth, smooth, sinuous with stable directionality, sinuous with variable directionality, and very sinuous. A three-layer back-propagation neural network (BPNN) composed of a fourteen-unit input layer, a three-unit hidden layer, and a five-unit output layer was also used. The quality of the results was analyzed by means of error matrices after a cross-validation process, which produced a goodness, or percentage of agreement, of greater than 83%. That study relates to ours with respect to its attempt to classify objects on the map by a machine learning method. Furthermore, map matching studies have been conducted to utilize machine learning. Hashemi and Karimi [11] used a two-layer feed-forward ANN for map matching. They applied ANN to decrease the horizontal error before matching Global Positioning System (GPS) points. ANN was trained through map-matched points and then used for correcting raw GPS points. They were then inputted into the map-matching algorithm. They tested their ANN with 10 to 20 hidden units. Two ANNs were used; one for estimating the length of the horizontal error, and the other for estimating the direction. Another study used two types of neural networks. Zhou and Li [12] applied a self-organizing map (SOM) and BPNN to road network data to update them. SOM worked as an unsupervised, clustering approach, while BPNN worked as a supervised classification approach. The results of the selective omission were then evaluated based on the overall accuracy. Moreover, in a recent study of Karsznia and Weibel [13], data enrichment and machine learning were used to improve the settlement selection for a small-scale map. They strived to identify a new map generalization rule using machine learning. A similarity existed with our study in terms of the applied technique; however, the primary focal points of the respective studies differed. Unlike our study, Karsznia and Weibel focused on deriving a new rule of map generalization. A difference also existed with respect to the data intended to be classified.

Although many studies have been conducted using machine learning, the approach has disadvantages. The algorithm performance greatly varies depending on the data quality. Therefore, it is necessary to obtain a large amount of high-quality data for the given research. Furthermore, the explanatory power of the algorithm results may not be as strong as those of conventional algorithms. Nevertheless, the use of machine learning should be considered because of its advantages of quickly processing large amounts of data and ensuring high accuracy.

1.3. Motivation for Building Classification

For the experimental data of the present study, we employed the building layer of a digital map. Buildings are one of the major geographical features of a map, and many previous studies on map generalization have focused on buildings. From among various map generalization operators, we focused on elimination and aggregation, which involve retaining or merging more important objects. Each building in the 1:1000 scale was labeled by the overlapping area with each building in the 1:25,000 scale. Then, the buildings were classified using representative classification algorithms of machine learning: decision tree (DT), *k*-nearest neighbor (*k*-NN), naive Bayes (NB), and support vector machine (SVM). In addition, geometric, topological, and thematic properties were used as input features. Our approach did not concern the whole process of building generalization; rather, it considered building classification, which could be used as a preparatory step for building generalization.

2. Study Area and Data Process

Figure 1 shows the workflow of this study. As the source data for testing, the building layer of a digital map with a scale of 1:1000 and 1:25,000, downloaded from the National Geographic Information Institute (NGII), was used [14]. Downloadable map datasets from NGII are available in five scales: 1:1000; 1:5000; 1:25,000; 1:50,000; and 1:250,000. The primary goal in this study was to update the 1:25,000 scale. We selected 1:1000 and 1:25,000 data as source data because these datasets showed more prominent differences than the other data sets. In total, 29,274 buildings of the Dongjak and Gwanaks districts were used for the test data (Figure 2). Both areas included a residential area and a central business district. The building densities were relatively high in these areas, and the building shapes had various forms, ranging from houses to large apartments and schools. Since the update frequencies of 1:1000 and 1:25,000 were different, 1:1000 data acquired in 2015 were used for data consistency.

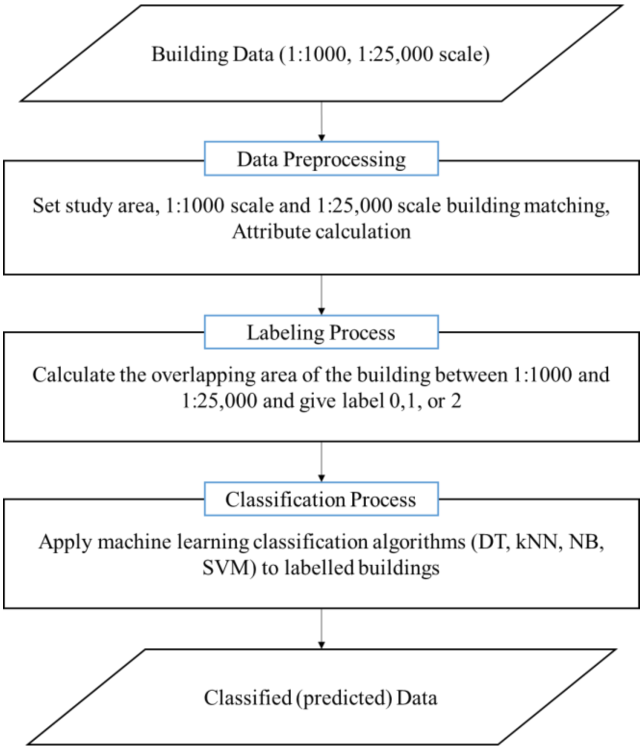

Building Data (1:1000, 1:25,000 scale)
Data Preprocessing
Set study area, 1:1000 scale and 1:25,000 scale building matching, Attribute calculation
Labeling Process
Calculate the overlapping area of the building between 1:1000 and 1:25,000 and give label 0,1, or 2
Classification Process
Apply machine learning classification algorithms (DT, kNN, NB, SVM) to labelled buildings
Classified (predicted) Data

Figure 1. Study workflow.

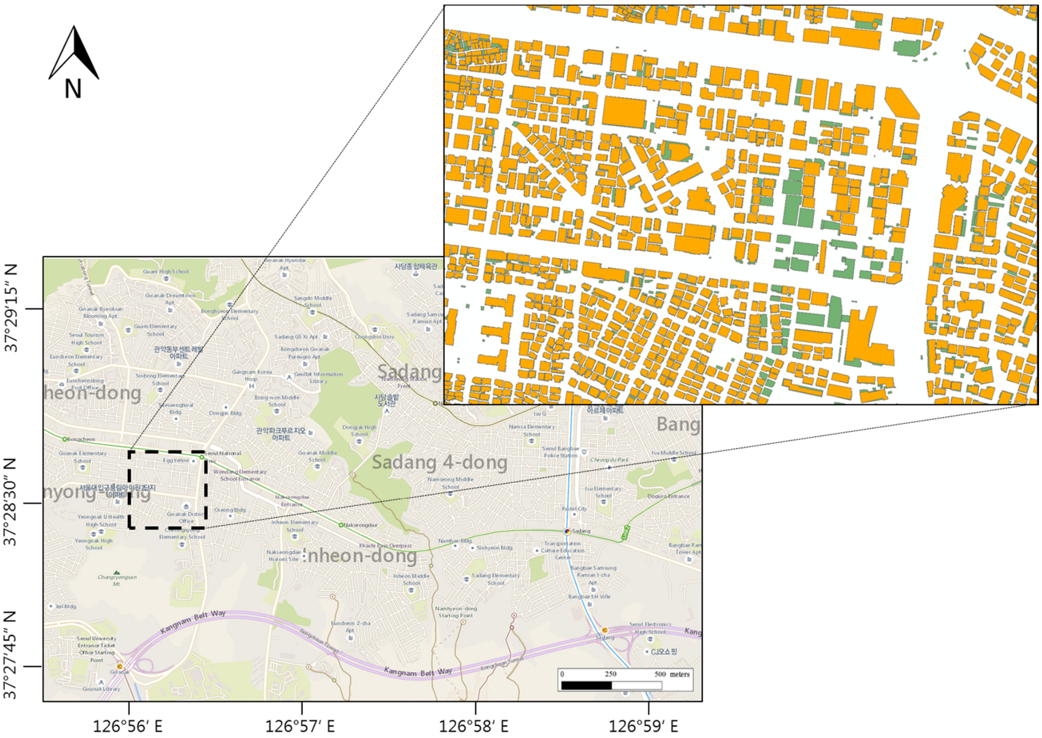


Figure 2. Test area represented in a Bing map. The magnified inset is the area where the changes are relevant between 1:1000 and 1:25,000 scales. Green = 1:1000 scale; orange = 1:25,000 scale.

The 1:25,000 and 1:1000 digital maps were rendered in the Drawing Exchange Format (.dxf) and Shape (.shp) format, respectively. For efficient data processing, it was necessary to unify these two data formats. Thus, the 1:25,000 building layer was converted into the Shape format. Next, it was necessary to extract the features that were necessary for classifying buildings. However, the digital map building layer did not contain all the required attributes, such as the building areas and perimeters (Table 1). Therefore, we separately calculated those attributes.

Table 1. Example of 1:1000 digital map building layer attributes.

Category	Apartment, town house, places of worship, etc.
Usage	Office, residential, commercial, education, etc.
Annotation	Annotation (building name)
Floor	Number of building floor
Address	Building address

After calculating the necessary attributes, the buildings were labeled as "0-eliminated," "1-retained," or "2-aggregated" according to whether they were eliminated or aggregated. These building attributes were used as an input feature, and the labeled values were used as a class for classification in the machine learning algorithm. After completion of the data preparation, the machine learning algorithms—DT, *k*-NN, NB, and SVM—were applied for classification.

3. Building Generalization and Classification

3.1. Overall Process of Building Generalization

Generalization of building data is usually performed in two distinct phases in most cartographic production workflows. The first is model generalization, which derives the scale-specific dataset from the master dataset. The main operator of model generalization is selective omission, which reduces the density of map features at the specific map scale. Other operators for map generalization are not considered in this phase. The second phase is cartographic generalization, which derives a data portrayal from the scale-specific dataset. In this phase, cartographic operators, such as aggregation, collapse, typification, exaggeration, line simplification, squaring, and displacement, are mainly applied.

Typically, aggregation, collapse, typification, and exaggeration are first applied to the dataset. Then, simplification and squaring are applied to the result obtained from applying the above operators. From that point, displacement is applied only when the results of the above operators cause a symbolic conflict between the map features, such as overlapping features, excessively high density, and so on. Since buildings are often densely situated in built-up areas or along linear features, such as roads, railroad tracks, rivers, and shores, generalization of building data and that of other feature classes affect each other. Cartographic operators of building generalization can be re-applied to the generalized results for reducing and eliminating conflicts caused by generalization of other feature classes.

3.2. Building Classification

In this study, we mainly focused on two operators, elimination and aggregation, and we had to generate training data to apply the machine learning technique to this process. Training data are required for applying classification algorithms, and they must contain input features and output classes. We strived to extract the input features from attributes of 1:1000 data and to generate the output class by comparing the areas of 1:1000 and 1:25,000 data (Figure 3). For our study, geometric, topological, and thematic attributes were used as input features. We used three geometric attributes: building area, perimeter, and height. As the height was not defined in the source data, we calculated it based on floor numbers.

The topological attributes were used as terms that implied a relation between buildings used in this study. A building distance with its closest building and the number of buildings in the threshold area were used as topological features. Moreover, the thematic attributes included building annotations and building functions, such as business, commercial, or residential. We transformed the annotation field into a binary attribute, that is, zero for no annotation and one for a building with an annotation. The building usage field was transformed into a categorical value.

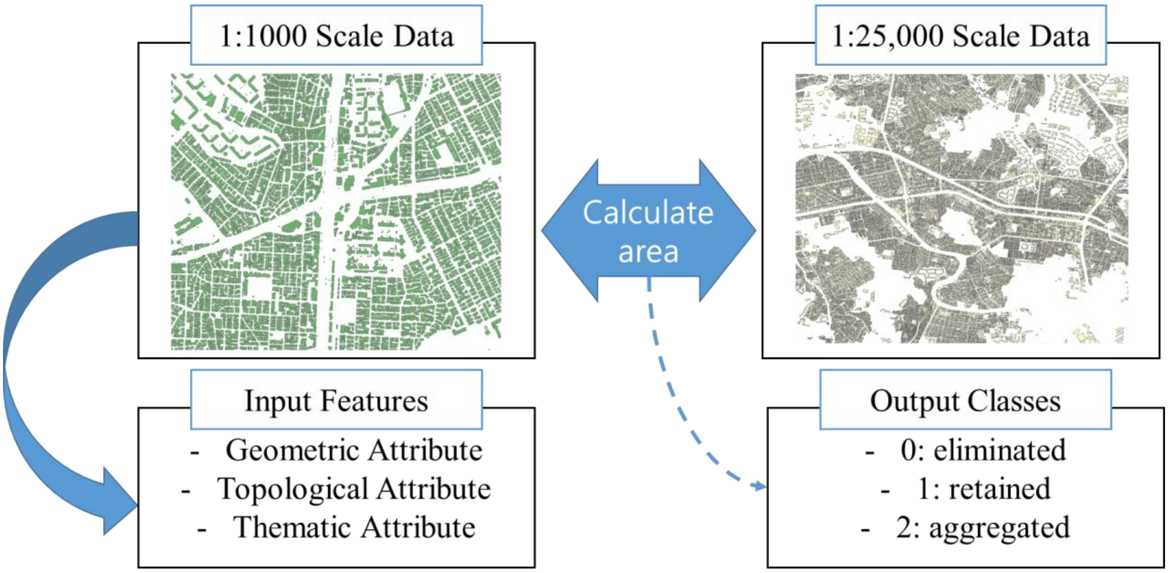


Figure 3. Schematic of the building classification.

For the representation of each building in a smaller scale, the following attributes were used as output classes: “0-eliminated,” “1-retained,” and “2-aggregated” (Figure 4).

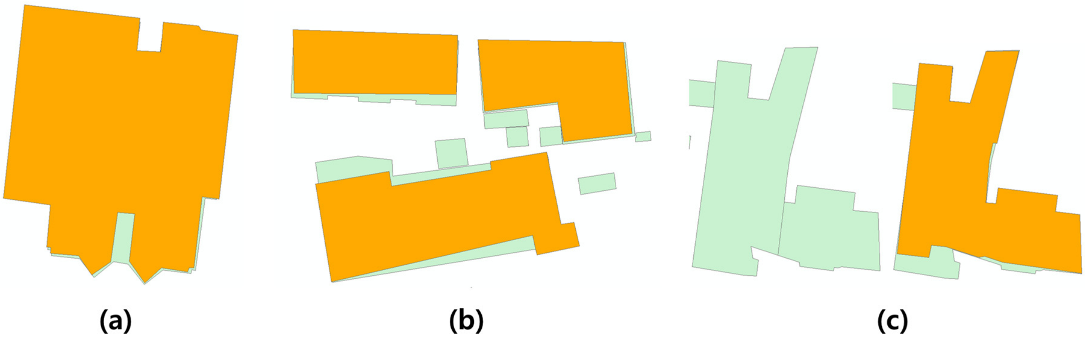


Figure 4. Examples of labeled buildings: (**a**) 0-eliminated; (**b**) 1-retained; (**c**) 2-aggregated (green = 1:1000 scale; orange = 1:25,000 scale).

The labeling was performed using the overlapping areas between 1:1000 and 1:25,000 buildings. The criterion for identification of two building objects in both datasets as representing the same building object in reality was the overlapping area, which was over 80% [15,16]. Figure 5 depicts the process for labeling the buildings.

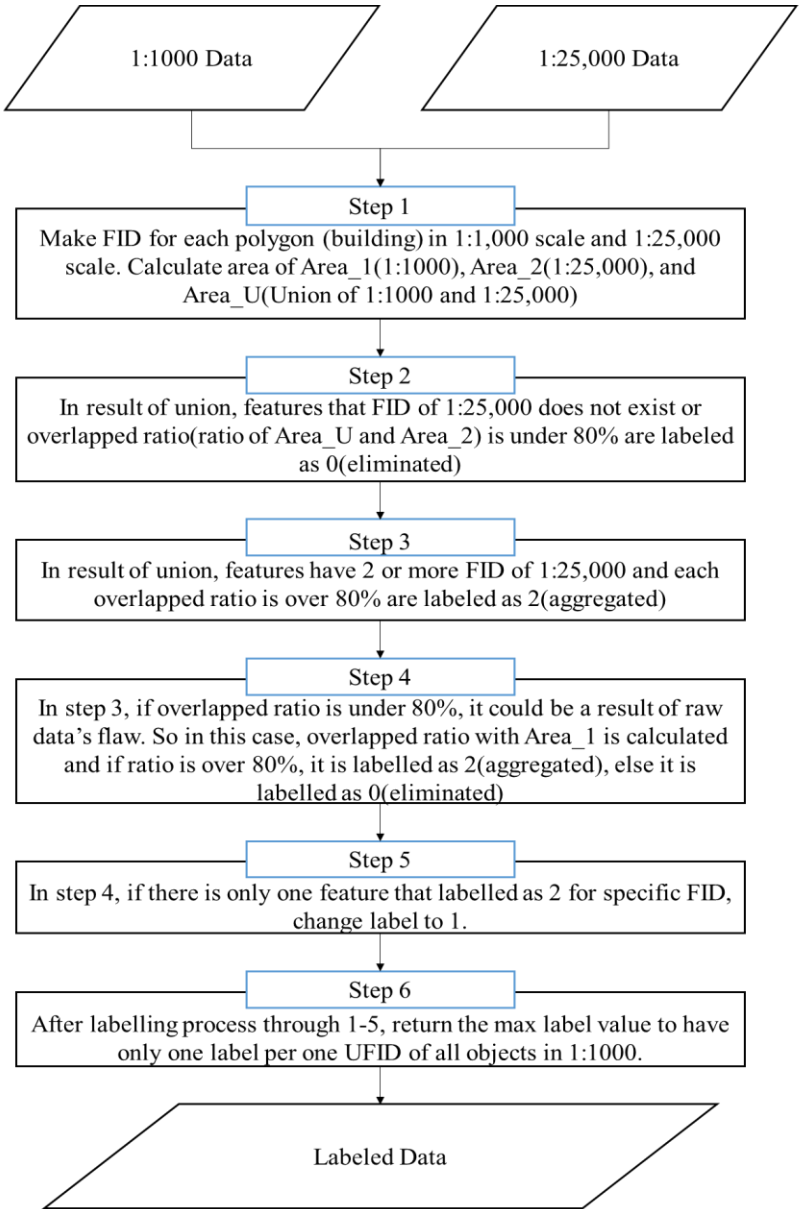

1:1000 Data
1:25,000 Data
Step 1
Make FID for each polygon (building) in 1:1,000 scale and 1:25,000 scale. Calculate area of Area_1(1:1000), Area_2(1:25,000), and Area_U(Union of 1:1000 and 1:25,000)
Step 2
In result of union, features that FID of 1:25,000 does not exist or overlapped ratio(ratio of Area_U and Area_2) is under 80% are labeled as 0(eliminated)
Step 3
In result of union, features have 2 or more FID of 1:25,000 and each overlapped ratio is over 80% are labeled as 2(aggregated)
Step 4
In step 3, if overlapped ratio is under 80%, it could be a result of raw data's flaw. So in this case, overlapped ratio with Area_1 is calculated and if ratio is over 80%, it is labelled as 2(aggregated), else it is labelled as 0(eliminated)
Step 5
In step 4, if there is only one feature that labelled as 2 for specific FID, change label to 1.
Step 6
After labelling process through 1-5, return the max label value to have only one label per one UFID of all objects in 1:1000.
Labeled Data

Figure 5. Building classification process.

4. Machine Learning Approach

In our study, as mentioned earlier, we applied DT, *k*-NN, NB, and SVM as the machine learning algorithms known to be suitable for classification. The descriptions of these algorithms are provided below.

4.1. Decision Tree (DT)

DT is a nonparametric supervised learning method that is used for classification and regression. Its goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. DT is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs arising from a node labeled with an input feature are labeled with each of the possible values of the target, output feature, or arc leading to a subordinate decision node on a different input feature. Each tree leaf is labeled with a class or a probability distribution over the classes (Figure 6).

The greatest advantage of DT is that it can easily and quickly classify target variables with very complex relationships by using a few simple input variables. Moreover, it is easy to interpret and understand the results, and qualitative and quantitative variables can be used without data preprocessing. Both classification and regression can be used, and various algorithms, such as single tree, random forest, and boost tree, were developed based on DT. The DT learning process is performed to maximize the information gain when adjusting input variables and boundaries. Accordingly, the information gain can be measured through entropy, which can be defined as [17]:

$$Entropy(S) = \sum(-p^{+}log_{2}p^{+} - p^{-}log_{2}p^{-}), \quad (1)$$

where p^{+} is the number of positive data and p^{-} is the number of negative data.

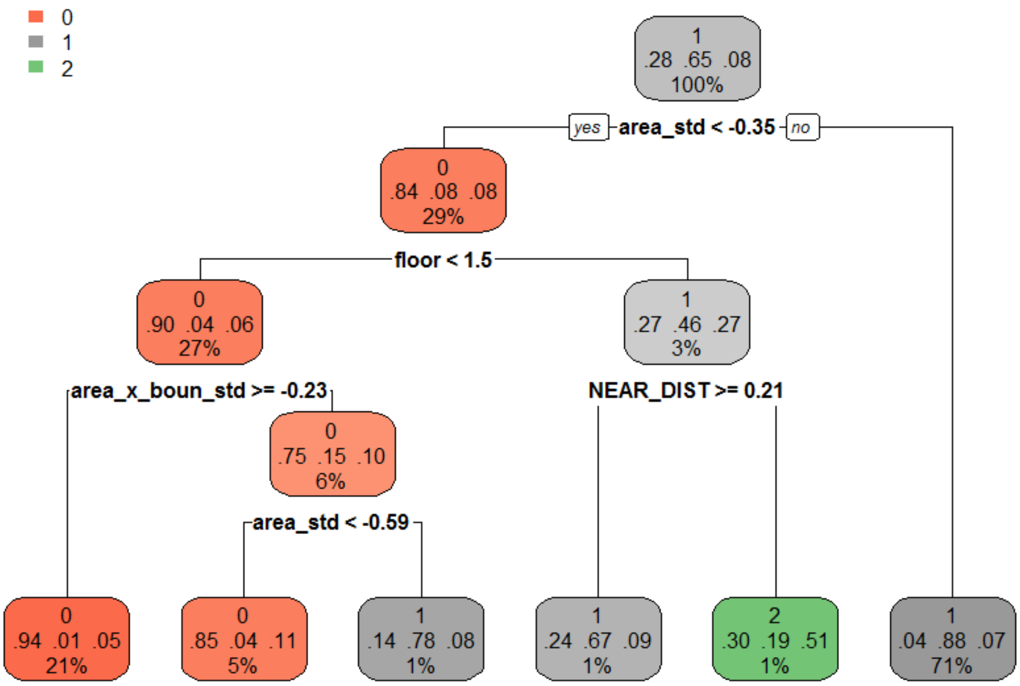


Figure 6. Decision tree of building classification.

The decision tree in this study was designed as follows: complexity parameter (CP) = 0.004, Minsplit = 4, and Minbucket = 2. Here, Minsplit means the minimum number of observations that must exist in a node for a split to be attempted. Minbucket means the minimum number of observations in any terminal leaf node. These values were experimentally determined.

4.2. *k*-Nearest Neighbor (*k*-NN)

K-NN is a simple supervised learning algorithm. When a data item is given, *k*-NN grasps the distance between a previously learned data item in the feature space and classifies it into the same item as the closest *k* data. The distance between variables is called the Euclidean or Manhattan distance. When *k* = 1, the search is the same as the simple nearest neighbor search. The advantage of *k*-NN is that it has a very fast processing speed with minimum data, few learning constraints, and a simple learning technique for very complex problems without setting any parameters.

However, if the amount of data is large, the processing time increases. Similar to other machine learning algorithms, *k*-NN is widely used in various fields, such as optical character recognition and genetic engineering. In this study, we used *k*max (maximum number of *k*) as 50, and the distance (Minkowski distance) was two. These values were experimentally determined.

4.3. Naive Bayes (NB)

NB is a map learning method by which classification is performed. It assumes that all input variables independently contribute to a variable to be predicted. For a given x^i, the probability that x^i belongs to C_k can be represented as Equation (2). The probability is calculated through Equations (3) and (4), and then it is classified as a class with the highest probability [18].

$$p\left(C_k|x^i\right) = p(C_k)\prod_{i=1}^{n} p\left(x^i|C_k\right) \tag{2}$$

$$p(C_k) = \frac{count(C_k)}{N} \tag{3}$$

$$p\left(x^i|C_k\right) = \frac{count\left(x^i, C_k\right)}{\sum_{x\in V} count\left(x^i, C_k\right)} \tag{4}$$

NB is characterized by training based on theoretical principles rather than training of a single algorithm. It can be applied with minimal data. NB is mainly used for classification of documents using spam filters and keywords. Moreover, it has a classification performance comparable to that of SVM. In the NB algorithm, we can adjust the parameters relating to Laplacian smoothing. In this study, we did not apply Laplacian smoothing because it was difficult to experimentally find an appropriate value.

4.4. Support Vector Machine (SVM)

SVM is a supervised learning method that is used in various fields, such as image classification, chemistry, and handwriting recognition. It has the best classification performance among the currently developed machine learning methods. It finds a hyperplane that can best classify data located in a vector space and classifies the data by it. The hyperplane has a boundary shape in the vector space. Its learning process works at maximizing the distance between the hyperplane and nearest data. Once the hyperplane is determined, the learned model is very efficient because most of the data used, except a minute amount, are no longer needed for prediction. When the data cannot be linearly classified, a method is required for expanding the data to a higher dimension. Since the calculation cost must be increased, a method for introducing a kernel was developed, thus greatly improving the classification performance of nonlinear data. In this study, we used linear SVM, and the cost was set to ten.

The advantages and drawbacks of each algorithm are identified as five items in Table 2.

Table 2. Advantages and drawbacks of each algorithm.

Algorithms	Accuracy	Training Speed	Result Interpretation	Versatility	Parameter Tuning Requirements
DT	Lower	Fast	Very clear	Classification Regression	Several
***k*-NN**	Lower	Fast	Very clear	Classification Regression	Minimal
NB	Lower	Fast	Somewhat	Classification only	Several
SVM	Higher	Fast (depending on the number of features)	Unclear	Classification Regression	Many

5. Experimental Results and Discussion

To apply the machine learning algorithm, we constructed training data consisting of seven input feature classes and three output classes. The input features included the annotation (binary), number of floors, building area, building perimeter, distance to the closest building, building category, and building complexity. The three output classes, as mentioned above, were 0-eliminated; 1-retained; and 2-aggregated. To find the required or relevant feature for machine learning, we calculated the information gain (IG) for our data. As a result, the area was determined to be the most important feature, while annotation was the least important feature. However, since our study did not have many input features, all input features were used in all the algorithms. The field values had to be standardized because the data deviation was considerable. The data standardization proceeded as follows:

$$x'_j = \frac{x_j - \mu_j}{\sigma_j}. \tag{5}$$

where μ is the mean of the field values, and σ is the standard deviation of the field values.

In this study, 70% of all data was used as training data, 10% was used as validation data, and 20% was test data. However, in our raw data, classes 0 and 1 corresponding to the eliminated and retained data, respectively, constituted the majority, and class 2 corresponding to the aggregated data was relatively less. Of the 29,274 buildings in the experimental area, only 2070 buildings were class 2 buildings.

The four types of machine learning classification algorithms were applied to the final experimental dataset by using the above process. Esri (Redlands, CA, USA)'s ArcMap 10.2 was used for data preparation, and R 3.3.0 was used for the machine learning applications. The number of correctly classified buildings was the output obtained from using the respective machine learning classification algorithms.

In the machine learning field, the results are usually represented by a confusion matrix (error matrix) [19], which is a specific table layout that enables visualization of the algorithm performance, typically that of a supervised learning algorithm. The confusion matrices were built based on test data comprising 20% of all the data, as mentioned above. The matrix for the machine learning algorithms is shown in Table 3.

Table 3. Confusion matrix for the test data.

			Predicted		
			0-Eliminated	**1-Retained**	**2-Aggregated**
DT	Actual Class	0-Eliminated	1454	141	2
		1-Retained	83	3645	0
		2-Aggregated	114	314	9
k-NN	Actual Class	0-Eliminated	1459	135	3
		1-Retained	110	3616	2
		2-Aggregated	109	317	11
NB	Actual Class	0-Eliminated	1513	78	6
		1-Retained	602	3054	72
		2-Aggregated	195	228	14
SVM	Actual Class	0-Eliminated	468	129	0
		1-Retained	150	3578	0
		2-Aggregated	123	314	0

Based on the above confusion matrix, an F-measure (F1-score), which evaluates the classification accuracy, could also be calculated. The F-measure is the harmonic mean of precision and recall. Precision is the number of correct positive results divided by the number of all positive results, and recall is the number of correct positive results divided by the number of positive results that should have been returned. The F-measure can be calculated as:

$$F_1 = \frac{2 * Recall * Precision}{Recall + Precision}. \tag{6}$$

In the case of multi-class classification, such as in our study, two indicators—micro-averaged F-measure and macro-averaged F-measure—are widely used. In micro-averaging, F-measure is computed globally over all category decisions. Micro-averaged F-measure can be calculated as:

$$F_{micro} = \frac{2 * Recall_{micro} * Precision_{micro}}{Recall_{micro} + Precision_{micro}}. \tag{7}$$

The micro-averaged F-measure gives equal weight to each document and is, therefore, considered an average over all the document/category pairs. It tends to be dominated by the classifier's performance on common categories.

In macro-averaging, the F-measure is first locally computed over each category, and then the average of all categories is obtained. The macro-averaged F-measure is obtained by taking the average of F-measure values for each category as:

$$F_{macro} = \frac{\sum_{i=1}^{M} F_i}{M} \tag{8}$$

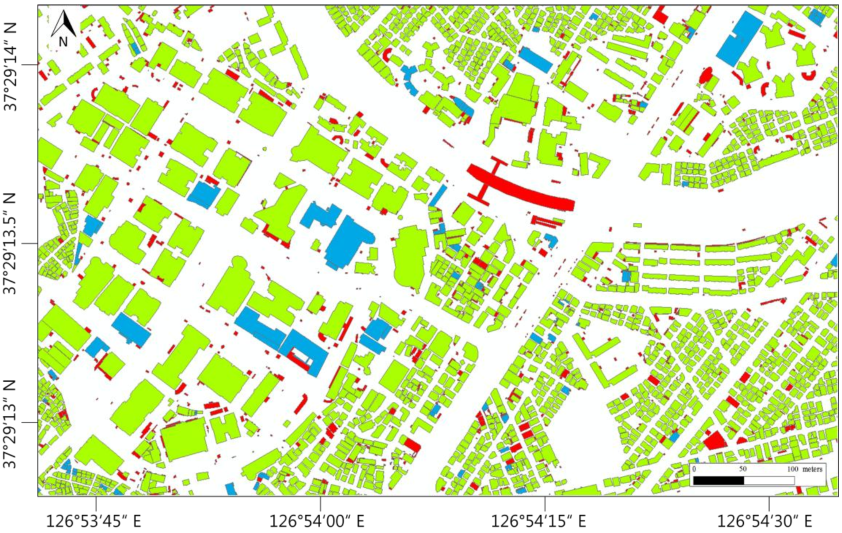
where M is total number of categories, and i is each category. Macro-averaged F-measure gives equal weight to each category, regardless of its frequency. It is influenced more by the classifier's performance on rare categories. We provide both measurement scores in Table 4 to be more informative.

Table 4. Macro and micro F-measures for test data.

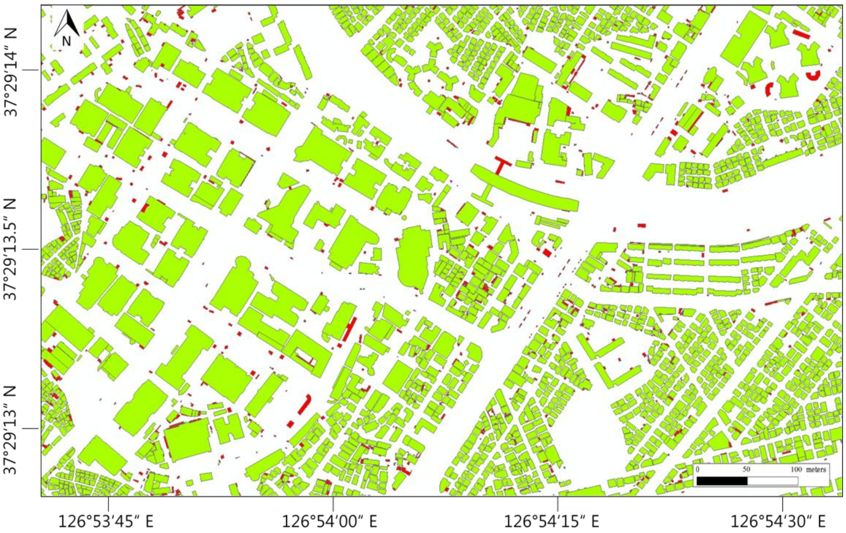
	Macro Average	Micro Average
DT	0.6223	0.8865
k-NN	0.6224	0.8825
NB	0.5631	0.7950
SVM	0.6009	0.8757

The overall accuracies (sum of the diagonal axis/total data from Table 3) of each algorithm were: DT, 88.96%; *k*-NN, 88.27%; NB, 79.50%; and SVM, 87.57%. NB showed significantly lower accuracy than the other three algorithms. This finding may likely be due to the assumption of independence, which is one of the characteristics of the NB algorithm. In all four algorithms, including NB, classes 0 and 1 performed the prediction with high accuracy, whereas class 2 did not effectively perform prediction. This is because the number of class 2 items in the learned model was insufficient, and the input features seemed to have had insufficient explanatory power for class 2.

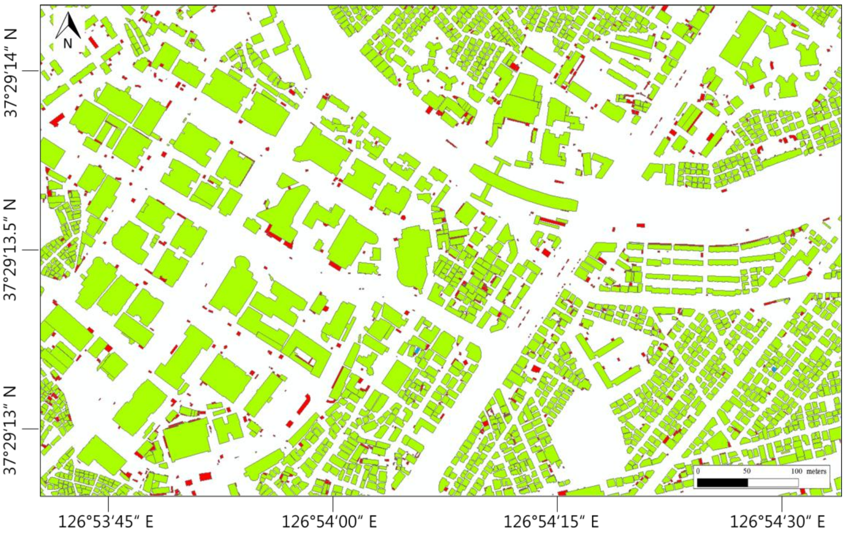
The results of the macro and micro F-measure also showed similar trends in overall accuracy. Classes 0 and 1 involved the elimination of buildings, whereas class 2 involved aggregation. Elimination was more intuitive and predictable than the other map generalization operators; however, aggregation seemed to have more difficulty classifying than our current machine learning model. The classification results of the respective algorithms are shown in Figure 7.



(**a**)

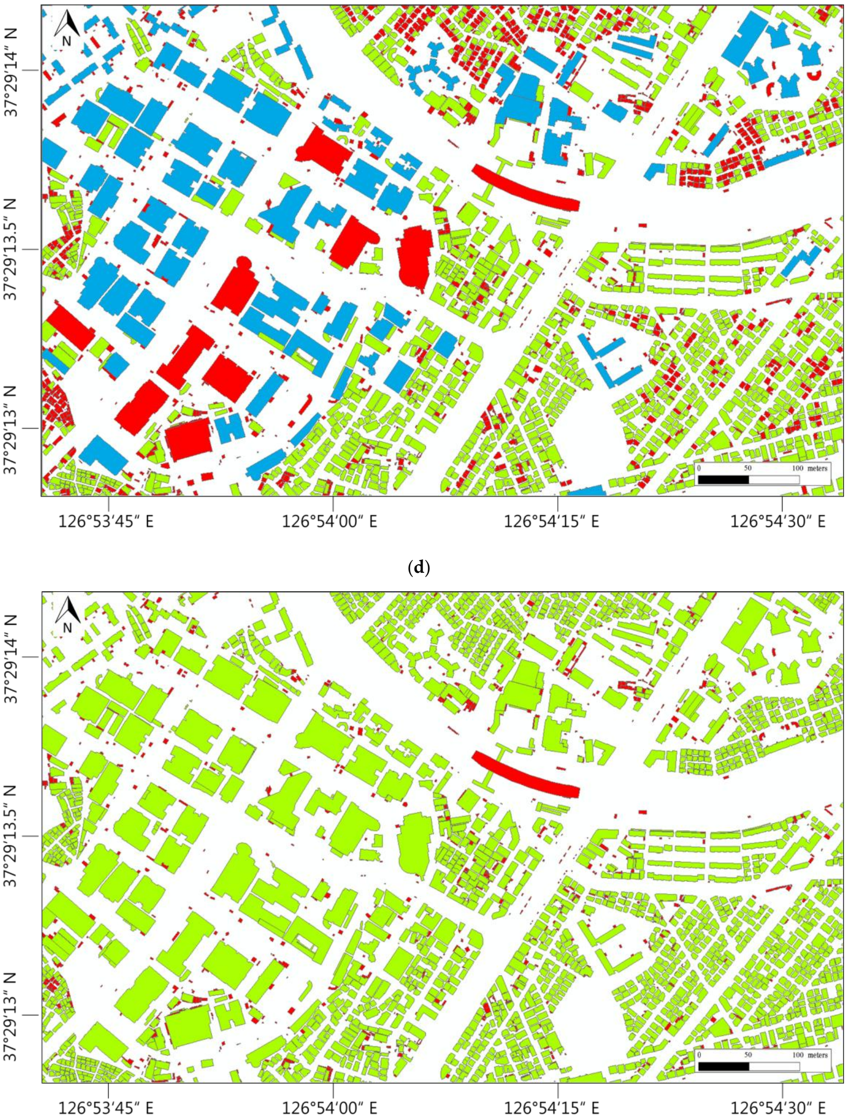


(**b**)



(**c**)

Figure 7. *Cont*.



(**d**)

(**e**)

Figure 7. Visualized results of each algorithm: (**a**) original class, (**b**) DT, (**c**) *k*-NN, (**d**) NB, and (**e**) SVM. Blue = eliminated (0); green = retained (1); and red = aggregated (2).

6. Conclusions

In this study, we applied machine learning algorithms to elimination and aggregation techniques of building generalization. To this end, buildings were labeled by determining whether the buildings of a 1:1000 scale were eliminated or aggregated at the 1:25,000 scale. We then applied the machine learning algorithm to a building by using the labeled values as the output class and the various property information of the building. The accuracy of each algorithm was also evaluated. To increase the accuracy of the aggregation case, it seemed necessary to either reinforce the proposed method itself or to apply another model in the case of aggregation. Our study showed that machine learning techniques can be applied to the preparation steps of building generalization. The research shows that a data-driven approach is possible, which is different from the existing complex mathematical and geometric methods.

In sum, the proposed technique can be used when there are multi-scale data (e.g., 1:5000 and 1:50,000 scale). It is simple to apply this method to other datasets, as long as the building is labeled by the proposed method or other methods. Results can be obtained by applying the above-noted machine learning algorithms, the output class resulting from labeling, and the attribute as the input feature, which will affect the output class. It should be noted that the data quality must be checked in applying the machine learning technique. If temporal accuracy, logical consistency, semantic accuracy, and completeness are not guaranteed, incorrect classification results can be obtained.

This study had some limitations. For one, other factors, such building patterns or map scales, were not considered. Furthermore, additional experiments are needed for various cases to consider various input properties in machine learning. In addition, the application to more building generalization cases should be addressed. We applied machine learning algorithms to only the elimination and aggregation techniques; however, it must be applied to other cases, such as displacement and simplification. Moreover, automation of the labeling process should be considered. It may be necessary to automate the process of comparing and labeling buildings of different scales. The application of machine learning techniques is also worth considering. Furthermore, to fully automate the building generalization process, the prediction accuracy must be higher. The prediction accuracy should be improved, and the application of other techniques, such as deep learning, should thus also be considered.

Acknowledgments: This research, "Geospatial Big Data Management, Analysis and Service Platform Technology Development," was supported by the MOLIT (The Ministry of Land, Infrastructure and Transport), Korea, under the national spatial information research program supervised by the KAIA (Korea Agency for Infrastructure Technology Advancement) (17NSIP-B081023-04).

Author Contributions: Jaeeun Lee and Kiyun Yu provided the core idea for this study. Hanme Jang and Jonghyeon Yang performed the data preprocessing and conducted some of the experiments. Kiyun Yu reviewed the manuscript, while Jaeeun Lee implemented the methodology and performed the experimental validation. Jaeeun Lee also wrote the main manuscript.

Conflicts of Interest: The authors have no conflicts of interest to declare.

References

1. Regnauld, N. Recognition of building clusters for generalization. In Proceedings of the 7th International Symposium on Spatial Data Handling, Delft, The Netherlands, 12–16 August 1996; pp. 185–198.
2. Anders, K.H.; Sester, M. Parameter-free cluster detection in spatial databases and its application to typification. *Int. Arch. Photogramm. Remote Sens.* **2000**, *33*, 75–83.
3. Ware, J.M.; Jones, C.B. Conflict reduction in map generalization using iterative improvement. *GeoInformatica* **1998**, *2*, 383–407. [CrossRef]
4. Li, Z.; Yan, H.; Ai, T.; Chen, J. Automated building generalization based on urban morphology and Gestalt theory. *Int. J. Geogr. Inf. Sci.* **2004**, *18*, 513–534. [CrossRef]
5. Wang, P.; Doihara, T. Automatic generalization of roads and buildings. *Triangle* **2004**, *50*, 1–7.
6. Damen, J.; Kreveld, M.; Spaan, B. High quality building generalization by extending the morphological operators. In Proceedings of the 11th ICA Workshop on Generalisation and Multiple Representation, Moscow, Russia, 2–3 August 2007; p. 12.
7. Park, K.S. A Study on the Consecutive Renewal of Road and Building Information in the Multi-Scale Digital Maps. *J. Korean Soc. Surv. Geod. Photogramm. Cartogr.* **2011**, *29*, 21–28. [CrossRef]
8. Vetter, A.; Wigley, M.; Käuferle, D.; Gartner, G. The automatic generalisation of building polygons with arcgis standard tools based on the 1: 50,000 swiss national map series. In Proceedings of the 18th ICA Workshop on Generalisation and Multiple Representation, Rio de Janeiro, Brazil, 21 August 2015.
9. Samuel, A.L. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.* **2000**, *44*, 206–226. [CrossRef]
10. Balboa, J.L.G.; López, F.J.A. Generalization-oriented road line classification by means of an artificial neural network. *Geoinformatica* **2008**, *12*, 289–312. [CrossRef]
11. Hashemi, M.; Karimi, H.A. A Machine Learning Approach to Improve the Accuracy of GPS-Based Map-Matching Algorithms. In Proceedings of the Information Reuse and Integration (IRI) 2016 IEEE 17th International Conference, Pittsburgh, PA, USA, 28–30 July 2016; pp. 77–86.
12. Zhou, Q.; Li, Z. Use of artificial neural networks for selective omission in updating road networks. *Cartogr. J.* **2014**, *51*, 38–51.

13. Karsznia, I.; Weibel, R. Improving settlement selection for small-scale maps using data enrichment and machine learning. *Cartogr. Geogr. Inf. Sci.* **2017**, 1–17. [CrossRef]
14. National Geographic Information Institute. Available online: https://www.ngii.go.kr/ (accessed on 26 June 2017).
15. Park, S.A.; Yu, K.Y.; Park, W.J. Updating building data in digital topographic map based on matching and generation of update history record. *J. Korean Soc. Surv. Geod. Photogramm. Cartogr.* **2014**, *32*, 311–318. [CrossRef]
16. Kim, J.Y.; Huh, Y.; Yu, K.Y.; Kim, J.O. Calculation of a Threshold for Decision of Similar Features in Different Spatial Data Sets. *J. Korean Soc. Surv. Geod. Photogramm. Cartogr.* **2013**, *31*, 23–28. [CrossRef]
17. Wang, Q.R.; Suen, C.Y. Analysis and design of a decision tree based on entropy reduction and its application to large character set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *PAMI-6*, 406–417. [CrossRef]
18. Murty, M.N.; Devi, V.S. *Pattern Recognition: An Algorithmic Approach*, 1st ed.; Springer: London, UK, 2011; pp. 86–102.
19. Stehman, S.V. Selecting and interpreting measures of thematic classification accuracy. *Remote Sens. Environ.* **1997**, *62*, 77–89. [CrossRef]