Proceedings of the IEEE
International Conference on Information and Automation
Ningbo, China, August 2016

Segmentation-Based Classification for 3D Urban Point Clouds

Binbin Xiang, Jian Yao†, Xiaohu Lu, Li Li, Renping Xie
Computer Vision and Remote Sensing (CVRS) Lab, School of Remote Sensing and Information Engineering
Wuhan University, Wuhan, Hubei, P.R. China
†Email: jian.yao@whu.edu.cn Web: http://cvrs.whu.edu.cn/

***Abstract*— In this paper, we propose a unified classification framework for 3D urban point clouds. First of all, an efficient segmentation approach is utilized to segment 3D point clouds. For comparison, we employed two recently developed point clouds segmentation approaches. The first one is a region-growing-based segmentation algorithm by using robust saliency features and another one is a hierarchical-clustering-based algorithm. The former will produce large and complete segments while the latter will generate more small pieces of segments. Secondly, we extract a set of features from each segment for training and testing by using an effective classifier. For comparison, we employed two popularly used classifiers, Naïve Bayes (NB) classifier and support vector machine (SVM). Thirdly, the contextual constraints among objects is used to further refine the classification results based on segments via graph cuts optimization. Experimental results on 3D urban point clouds acquired by a vehicle LiDAR system illustrate that our proposed classification framework is effective.**

***Index Terms*— 3D Point Clouds, Segmentation, Classification, Graph Cuts Optimization, Urban Scene**

I. INTRODUCTION

Nowadays, since the intelligent mobile robot can efficiently and independently complete many rough tasks for us and take many convenience for our life and production, it has attracted a lot of attention and has been rapidly developed. For an intelligent robot, it must know itself where it is in an unknown environments before it makes the next decision and completes a specific task, such as self-driving and rescue missions. In addition, to ensure that the robot can complete the task in arbitrary environments, it also need to plan the optimal route for the task in current environment instead of just moving according to the previously designed route. In general, those problems can be formulated as Simultaneous location and mapping (SLAM) and Path Planning problems in the robotics society. The SLAM is a process of building and updating a map of unknown environment based on the data collected by the sensors mounted on the mobile robot platform and simultaneously determining itself where it is located on the map. Based on the constructed map, the robot also can guide itself intelligently via the path planning algorithm. However, for some tasks, it is not enough only to know the information of location, it also need to parse the neighbor environment. To better understand the unknown urban scenes and complete the complex tasks, the classification algorithm should be applied to identify some specific objects in the scenes and obtain their location information based on the results of positioning. As the rapid development of 3D data acquisition technology, especially 3D laser scanning, the 3D point clouds can be obtained easily and cheaply. The classification for 3D point clouds is a classical and important problem in the field of computer vision and robotic society, which is a key technology for scene understanding and detailed semantic analysis.

In last several decades, many methods are proposed to solve the point cloud classification problem. Generally, existing methods can be divided into two main categories according to the basic element employed in classification: point-based classification and segment-based classification. The point-based methods classify the 3D point clouds by analyzing the characteristics of single point. Shan and Aparajithan classified ground points from raw LiDAR data for bald ground digital elevation model (DEM) generation in urban areas by labeling single point as either a ground point or a non-ground point [1]. Guo et al found that the multiple returns and echoes of laser pulse can be discriminant features for single point used to distinguish adjacent objects [2]. However, those point-based methods maybe failed in some complex point classification conditions due to the features extracted in the scale of point are limited. Therefore, for complex classification consists of multiple types of objects, many methods segment the original point clouds at first, and then perform a classification based on those segments. These methods not only can improve the slow computational rate problem resulting from the increasing amount of point cloud data, but also extract more richer information than point-based methods. Thus, segment-based classification have drawn more attentions in recent years.

Point clouds segmentation is a key issue for classifying the point clouds, and many methods have proved that. The popularly used feature in point segmentation is surface discontinuities. For instance, the method presented in [3] used only local surface normals and point connectivity to segment the industrial point clouds and performs well. For urban scene, other surface features, such as normal, curvature and the height difference, were widely used to find the smoothly connected areas [4], [5]. In addition, many algorithms applied graph cuts [6], [7] and Markov random fields [8] to generate the more smoother segments via using neighborhood smoothing constraints. The basic ideal of those methods is first constructed a weighted graph where each edge weight

978-1-5090-4102-2/16/$31.00 ©2016 IEEE

cost represents the similarity of the corresponding segments, and then find the minimum solution in this graph. The k-Nearest Neighbors (k-NN) [9] algorithm is often used to build the graph to improve the efficiency.

After the segmentation, certain classification algorithm is employed to assign the segment with a class. Traditionally, point cloud classification is completed by manually defining a series of discriminant rules to distinguish points for each class. For example, we can assume that the ground have the minimum height in the neighboring regions according to our knowledges and use this constraint to label all points fitted this measurement as the ground. However, the rules for classification is difficult to set in many cases. To solve this problem, the machine learning method can be applied to learn the classification rules automatically from training data [10]. For example, in outdoor urban scenes, the researchers used SVM to distinguish basic categories, such as building, ground and vegetation [11], [12]. In addition, the random forests algorithm was also successfully applied to the LiDAR feature selection to classify urban scene in [13]. Moreover, the AdaBoost algorithm formed a strong classifier by using simple geometric features extracted from single laser range scans to classify the points into several semantic classes, like rooms, hallways, corridors, and doorways [14]. However, most of the common classifiers just take into account local features to complete the point classification and ignore the topological or geometric relationships between different objects which usually existed in urban environments, so it is a effective way to improve the accuracy of classification results by integrating the contextual information into the machine learning framework. Classification approach of a LiDAR point cloud based on Conditional Random Fields (CRF) successfully obtained three basic object classes: vegetation, building and ground [15]. Combining CRF with random forest classifier can obtain more reliable classification results, especially the number of confusions between building and larger trees reduced obviously [16]. Moreover, Associative Markov Network (AMN) was widely used to classify 3D point cloud by utilizing contextual information [17], [18].

Considering the advantages of both the segmentation-based classification and the context information, we proposed an effective algorithm for unstructured 3D point clouds classification. At first, two different kinds of point cloud segmentation approaches are applied to separately segment the origin point cloud data. Then, based on the segmentation results, different classifiers are utilized to classify the segments into several specific classes. And in our work, the applied classifiers are NB classifier and SVM. Last but not least, the classification results are optimized based on the graph cuts energy minimization algorithm. In the experiment section, we present the classification results of our method on challenging dataset and compare the results with different segmentation and classification methods used.

II. Our Approach

In our work, the overall work-flow can be separated into three steps. The overview flowchart is depicted in Fig. 1.

The first step is segmenting the original unstructured 3D point cloud. For comparison, we employed two different segmentation algorithms. The first one is a region-growing-based segmentation algorithm by using robust saliency features and another one is a hierarchical-clustering-based algorithm. Secondly, we extract a set of features from each segment for training and testing by using an effective classifier. For comparison, based on the segmentation results, NB classifier and SVM are applied respectively to classify segments. In the end, as the classifier cannot give hundred percent correct classification results, we use contextual constraints among objects via graph cuts energy minimization algorithm to optimize the initial classification.

A. Point Cloud Segmentation

1) RDPCA-Based Segmentation: Normal and curvature are two widely used saliency features in point cloud segmentation, which are calculated using conventional Principal Component Analysis (PCA). However, the PCA is very sensitive to noises and outliers which are quite common in the original point cloud data, the local saliency features obtained by the traditional PCA methods are usually not robust and inaccurate. In order to get robust results, a region growing based segmentation algorithm was proposed by using a Robust Diagnostic PCA (RDPCA) approach based on the Maximum Consistency with Minimum Distance (MCMD) [19].

The detailed implementation steps of the proposed RDPCA based segmentation method are as follows. Firstly, for each data point p_i, the robust saliency features, normals and curvatures of it is calculated by RDPCA [20] which uses the MCMD-Z based robust diagnostic technique to find outliers in the local neighborhood of p_i and then calculate PCA from the inlier points. Then, all the points are sort ascending according to their curvature. The foremost unprocessed point is selected as the seed point, then a region grows procedure is started from the seed point to collect neighboring points which are close to the seed point, iteratively. Three measurements are applied to indicate the closeness of a neighboring point p_i to the seed point p_{seed}, including: the orthogonal distance from p_i to fitted plane of p_{seed}, the Euclidean distance from p_i to p_{seed}, and angular distance between the two points.

2) P-Linkage-Based Segmentation: P-Linkage is a recently proposed hierarchical clustering algorithm, which segments the unstructured point cloud data based on clustering analysis [21].

The detailed implementation steps of the proposed P-Linkage based segmentation method are as follows. Firstly, for each data point, the normal, curvatures, and Consistent Set (CS) of it are obtained via both PCA and MCMD-Z

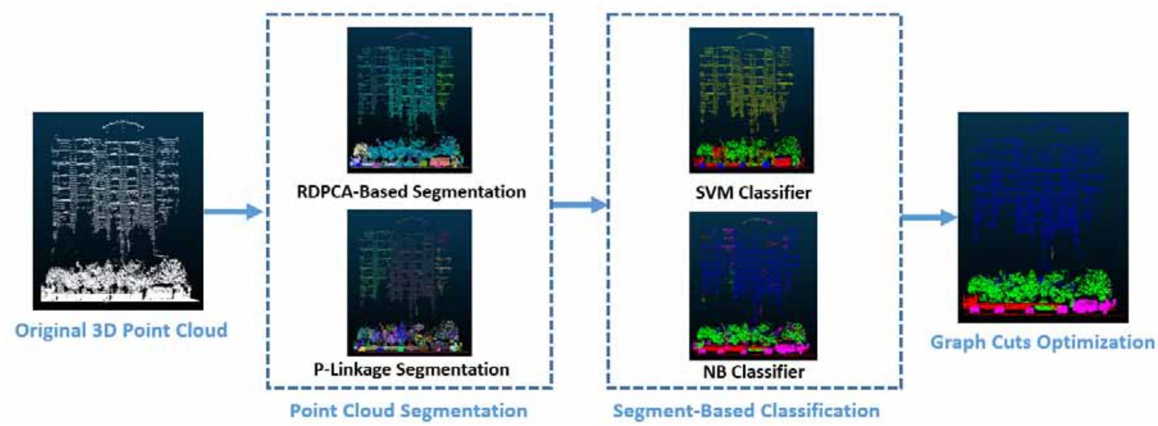


Fig. 1. Overview flowchart of the whole procedure.

algorithm. Then, the linkage between each data point p_i and its closest neighboring point (CNP) is built. The CNP is defined as the neighbor point of p_i who is the closest one to p_i among those neighbors with smaller curvatures than p_i. If p_i is the one with smallest curvatures in its neighborhood, we consider p_i as a cluster center. Thirdly, the initial clusters can be discovered by searching along the linkages started from each cluster center, each cluster corresponds to a segment. Finally, an efficient segment merging method is proposed to merge adjacent segments when the condition to be merged is satisfied .

B. Segment-Based Classification

1) Naïve Bayes Classifier: Classification by NB classifier only need to calculate the posterior probabilities for each classes of the inputting sample, and assign the class label with maximum posterior probabilities to it. Given sample S and the number of classes N_c, the posterior probabilities of the l-th class C_l can be formulated as below:

$$P(C_l \mid S) = \frac{P(S \mid C_l)P(C_l)}{P(S)}, \tag{1}$$

where $P(S \mid C_l)$ presents the likelihood, $P(C_l)$ is the prior probability of the l-th class, and $P(S)$ is the model evidence, calculated by $P(S) = \sum_{q=1}^{N_c} P(S \mid C_q)P(C_q)$. We assumed that the prior probability of the classes are equal, means that $P(C_l) = 1/N_c$.

For the purpose of classifying all unknown-class segments according to component difference and coordinate distribution of different types points of every segment, before the training step, we should operate some data preprocessing, including point type extraction and coordinate transformation. As shown in Algorithm 1, we calculate a 4-dimensional binary vector $\mathbf{z}_n$ to indicate the point type of each point by using the normals and eigenvalues calculated in segmentation process. As for coordinate transformation, firstly, we calculate $\mathbf{X}'_n = (X'_n, Y'_n, Z'_n)^\top$ by subtracting average vector from the global coordinate vector $\mathbf{X}_n = (X_n, Y_n, Z_n)^\top$ of one point in set $\mathcal{X} = \{\mathbf{X}_i\}_{i=1}^N$. Then, we transform X'-Y' axes into U-V axes, which are the principle axes obtained by the PCA. Finally, we define $\mathbf{x}_n = (U_n, Z'_n)^\top$ as coordinate vector for each point. For each segment, we can present it by using a set of $S = \{X, Z\}$, where X and Z represent a set of vectors indicating coordinates and point type of each point, respectively.

Algorithm 1 Point Type Extraction

Input: The normal $\vec{\mathbf{n}} = (n_x, n_y, n_z)^\top$, the minimum eigenvalue λ_0, and the maximum eigenvalue λ_2 of the current point p_i.
Output: The 4D vector $\mathbf{z}_n$ indicating the point type of the current point.

if $\lambda_0/\lambda_2 > 0.1$ **then**
  p_i is scatter type and $\mathbf{z}_n = (1, 0, 0, 0)^\top$.
else
  calculate the angle θ between $\vec{\mathbf{n}}$ and Z-axis.

$$\theta = \tan^{-1} \frac{|n_z|}{|n_x|^2 + |n_y|^2} \tag{2}$$

  if $\theta < 30^\circ$ **then**
    p_i is horizontal type and $\mathbf{z}_n = (0, 1, 0, 0)^\top$.
  else if $30^\circ \le \theta \le 60^\circ$ **then**
    p_i is slope type and $\mathbf{z}_n = (0, 0, 1, 0)^\top$.
  else if $60^\circ < \theta < 90^\circ$ **then**
    p_i is vertical type and $\mathbf{z}_n = (0, 0, 0, 1)^\top$.
  end if
end if

For each training example, the likelihood which represent the distributions of point clouds with point types can be modeled by Gaussian Mixture Model (GMM) [22]. Actually, there are more than one training examples for each class. In the training step, we can obtain all GMM parameters of all training examples. And the posterior probability for one of class can be obtained by summing all the posteriors in this class.

2) Support Vector Machine: SVM is a very popular classifier and has been used in many fields of computer vision, which is possible to split apart different types of samples in high-dimensional space by obtaining the most optimal

TABLE I

WEIGHTS OF EDGES IN GRAPH CUTS.

Edge	Weight	Condition
$\{p, q\}$	$W_{p,q}$	$\{p, q\} \in \mathcal{N}$
$\{p, s\}$	$W_{p,s}$	$p \in \mathcal{P}, p \notin \mathcal{F} \cup \mathcal{B}$
	$\max\left(W_{p,s}\right)$	$p \in \mathcal{F}$
	0	$p \in \mathcal{B}$
$\{p, t\}$	$W_{p,t}$	$p \in \mathcal{P}, p \notin \mathcal{F} \cup \mathcal{B}$
	0	$p \in \mathcal{F}$
	$\max\left(W_{p,t}\right)$	$p \in \mathcal{B}$

hyperplanes. And the accuracy of classification results is largely depend on the quality of the predefined features. In this work, according to the characteristics of different types of objects in urban scene, we define three kinds of geometrical and local descriptors. The first one is the average height of segment, which is a very useful feature to distinguish the objects with different height, like building, car and ground. Secondly, objects in different classes have different geometrical shape in general. For instance, the ground can be represented as a low flat plane, and the buildings can be represented as large, vertical blocks both in width and height. However, the minimum bounding box of trees and cars are almost broad and short. So, we calculate two kinds of projection area. One is the projection area in the X-Y axes plane, the other is the projection area in the U-Z axes plane according to previous description. Besides, the orientation of different classes is a kind of essential feature can be extracted from the segmentation results. At first, the normal vector of the entire segment can be obtained by PCA, and we calculate the angle between the normal and Z-axis as part of orientation information. Then, similar to the method used in NB classifier, we respectively calculate the number of points divided by total points number in the segment for 4 different types, as the other four orientation features for SVM classification.

C. Optimization via graph cuts

By observing the initial classification results, we find that there always exist many misclassified regions. To optimize the initial classification and achieve a more smooth and accurate result, we formulate this problem as an energy optimization problem and solve it via graph cuts algorithm, which merge the small misclassification with its nearest and reliable classification objects.

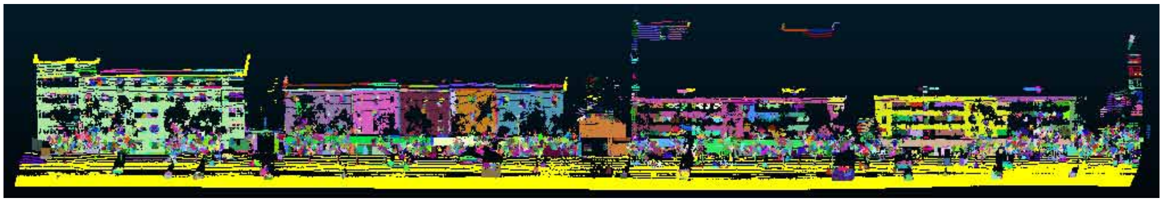
The region growing algorithm is firstly used to cluster nearby points with the same initial classification labels. We regard the objects with large number of clustered points as the reliable class, while objects with a tiny number of clustered points are regarded as unreliable class. And we simply set the threshold as the average or median number of points in segments for each class. Then for each reliable object, the graph cuts optimization is used as follow steps. The initial range of optimization is a spherical region which the center point is the gravity point and consider the maximum distance between center point and the point in current reliable object as the radius. Then the bin is marginally expanded until it contains not only current reliable object, but also contains points both in other reliable objects and unreliable region. For all points in current reliable object, we consider them as foreground seed points. And the points in other reliable objects in this region are considered as background seed points. Every points in the range of optimization are used to construct the graph [23]. The weights of edges connecting points at a distance d are calculated according to Table I, where $\mathcal{N}$ and $\mathcal{P}$ represent the set of edges and nodes in graph, and p, $q \in \mathcal{P}$. s and t are the average points in foreground set $\mathcal{F}$ and background set $\mathcal{B}$. $W_{p,q} = \exp\left(-\left(\frac{d}{\sigma}\right)^2\right)$, and σ is the average distance of points in the LiDAR dataset. Finally, we find the minimum solution via the min-cut algorithm by segmenting the points into the classes of foreground or background.

III. EXPERIMENTAL RESULTS

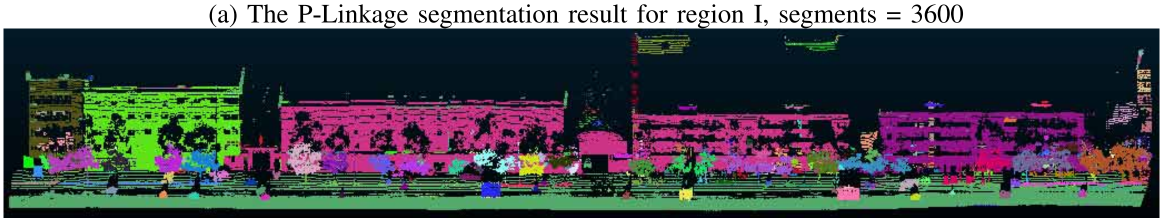
In this section, to prove that our proposed algorithm is effective, we tested our algorithm in two dataset, region I is an urban street of 241 meters long with 365608 points and region II is a small scene with total 31491 points. From the experiment results showed in Fig. 2, 3, 4, we can draw the following conclusions. Firstly, the segmentation based on RDPCA will produce large and complete segments while P-Linkage segmentation will generate more small pieces of segments. As for the two classifiers, the SVM obtain the similar results based on two different segmentation algorithm. However, the NB classifier can obtain a more better result based on object-based segments acquired by the segmentation based on RDPCA. In addition, the NB classifier have the advantage of training with few examples, but without considering the topological relations among the segments, it will lead to some ridiculous misclassification, such as sometimes the car is higher than the house as showed in Fig. 3-(c). In addition, we found that our proposed classification refinement method based on graph cuts is effective, as shown in Fig. 4, where some misclassification regions are clearly filtered out after optimization step. The confusion matrix for initial classification and after optimization of Region II are reported in Tables II and III, respectively. We observed that both the classification precision and recall with the optimization are higher than that without optimization.

IV. CONCLUSIONS

In this paper, we proposed a “segmentation-classification-optimization” work-flow to classify the point clouds captured from the urban scenes. Firstly, we employed two recently developed point clouds segmentation algorithms to obtain the segments and evaluated their influences for the last classification, respectively. The first one is a region-growing-based segmentation algorithm by using robust saliency features and



(a) The P-Linkage segmentation result for region I, segments = 3600



(b) The RDPCA based segmentation result for region I, segments = 291

Fig. 2. Segmentation results of the two different kinds of method for region I.

TABLE II

REGION II INITIAL NB CLASSIFICATION CONFUSION MATRIX.

	Ground	House	Tree	Car	Recall
Ground	2569	220	1232	381	0.584
House	0	9344	0	0	1.000
Tree	777	313	8035	27	0.878
Car	324	622	981	6666	0.778
Precision	0.700	0.891	0.783	0.942	
Overall Classification Precision: 0.829					

TABLE III

REGION II CONFUSION MATRIX FOR CLASSIFICATION .

	Ground	House	Tree	Car	Recall
Ground	3208	0	87	72	0.953
House	7	10473	0	0	0.999
Tree	454	0	10151	0	0.957
Car	1	16	20	7002	0.995
Precision	0.874	0.999	0.990	0.990	
Overall Classification Precision: 0.963					

another one is the P-Linkage segmentation. Secondly, the machine learning algorithm is used to learning the rules from the predefined training data. For comparison, two popularly used classifiers, NB classifier and SVM are applied. Finally, the contextual constraints among objects are integrated into the graph cuts energy optimization framework to further refine the classification results. Experimental results on challenging urban point clouds collected by the laser mounted on a vehicle shown that our proposed classification framework is effective.

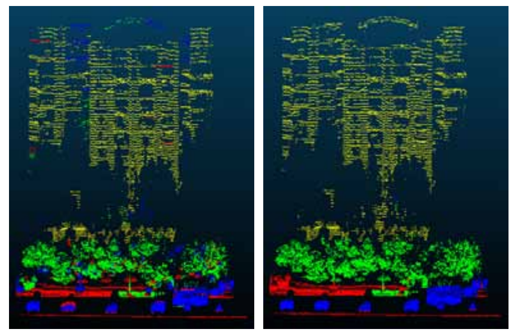


Fig. 4. The left figure shows the initial classification and right figure is the result after optimization for the region II data.

ACKNOWLEDGMENT

This work was partially supported by the National Natural Science Foundation of China (Project No. 41571436), the Hubei Province Science and Technology Support Program, China (Project No. 2015BAA027), the Jiangsu Province Science and Technology Support Program, China (Project No. BE2014866), and the South Wisdom Valley Innovative Research Team Program.

REFERENCES

[1] Sampath Aparajithan and Jie Shan, "Urban DEM generation from raw LiDAR data," *Photogrammetric Engineering and Remote Sensing*, vol. 71, no. 2, pp. 217–226, 2005.

[2] Li Guo, Nesrine Chehata, Clément Mallet, and Samia Boukir, "Relevance of airborne LiDAR and multispectral image data for urban scene classification using Random Forests," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 1, pp. 56–66, 2011.

[3] Tahir Rabbani, Frank Van Den Heuvel, and George Vosselmann, "Segmentation of point clouds using smoothness constraint," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, pp. 248–253, 2006.

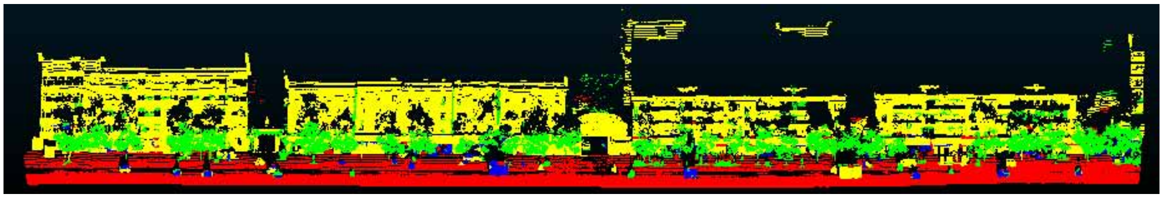
[4] Klaas Klasing, Daniel Althoff, Dirk Wollherr, and Martin Buss, "Comparison of surface normal estimation methods for range sensing applications," in *IEEE International Conference on Robotics and Automation*, 2009.

[5] David Belton and Derek D. Lichti, "Classification and segmentation of terrestrial laser scanner point clouds using local variance information," in *ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'*, 2006.

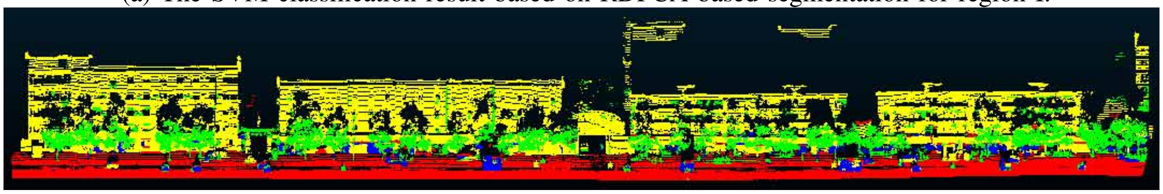
[6] Aleksey Golovinskiy and Thomas Funkhouser, "Min-cut based segmentation of point clouds," in *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2009.

[7] Johannes Strom, Andrew Richardson, and Edwin Olson, "Graph-based segmentation for colored 3D laser point clouds," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.

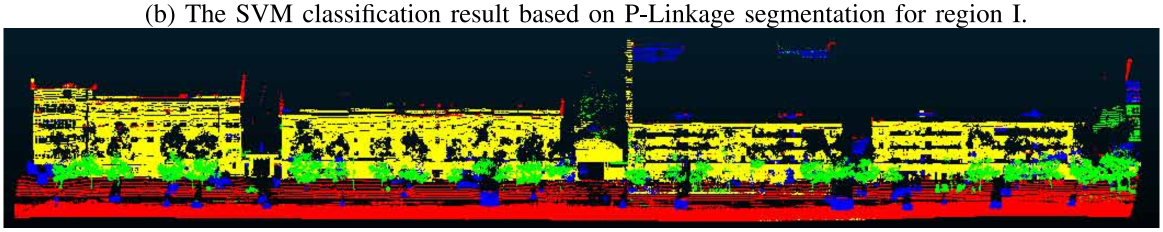
[8] Dragomir Anguelov, B Taskarf, Vassil Chatalbashev, Daphne Koller, Dinkar Gupta, Geremy Heitz, and Andrew Ng, "Discriminative learning of Markov random fields for segmentation of 3D scan data,"



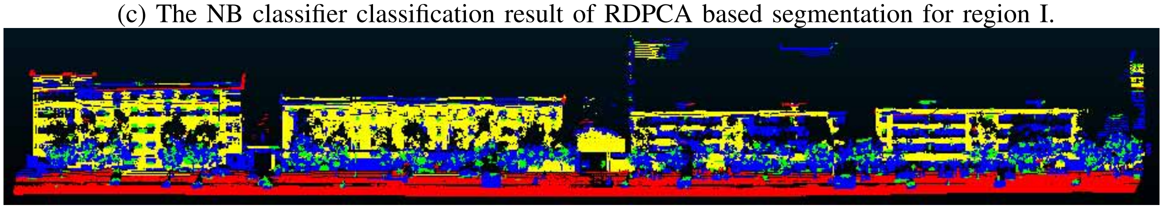
(a) The SVM classification result based on RDPCA based segmentation for region I.



(b) The SVM classification result based on P-Linkage segmentation for region I.



(c) The NB classifier classification result of RDPCA based segmentation for region I.



(d) The NB classifier classification result of P-Linkage segmentation for region I.

Fig. 3. Initial classification results for region I. Here the yellow color represents the building class, the green color represents the tree class, the blue color represents the car class and the red color represents the ground class respectively.

in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[9] Sunil Arya, David M Mount, Nathan S Netanyahu, Ruth Silverman, and Angela Y Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *Journal of the ACM*, vol. 45, no. 6, pp. 891–923, 1998.

[10] C.M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, New York, 2007.

[11] Jixian Zhang, Xiangguo Lin, and Xiaogang Ning, "SVM-based classification of segmented airborne LiDAR point clouds in urban areas," *Remote Sensing*, vol. 5, no. 8, pp. 3749–3775, 2013.

[12] Tao Tang and Lixian Dai, "Accuracy test of point-based and object-based urban building feature classification and extraction applying airborne LiDAR data," *Geocarto International*, vol. 29, no. 7, pp. 710–730, 2014.

[13] Nesrine Chehata, Li Guo, and Clement Mallet, "Airborne lidar feature selection for urban classification using random forests," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, pp. W8, 2009.

[14] Oscar Martinez Mozos, Cyrill Stachniss, and Wolfram Burgard, "Supervised learning of places from range data using AdaBoost," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[15] J. Niemeyer, C. Mallet, F. Rottensteiner, and U. Soergel, "Conditional random fields for the classification of LiDAR point clouds," in *ISPRS - International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2011.

[16] Joachim Niemeyer, Franz Rottensteiner, and Uwe Soergel, "Classification of urban LiDAR data using conditional random field and random forests," in *Joint Urban Remote Sensing Event (JURSE)*, 2013.

[17] Daniel Munoz, Nicolas Vandapel, and Martial Hebert, "Directional associative markov network for 3-D point cloud classification," in *Fourth International Symposium on 3D Data Processing, Visualization and Transmission*, 2008.

[18] Delfina Munoz, J Andrew Bagnell, Nicolas Vandapel, and Martial Hebert, "Contextual classification with functional max-margin markov networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[19] Abdul Nurunnabi, David Belton, and Geoff West, "Robust segmentation for large volumes od laser scanning 3D point cloud data," *IEEE Transactions on Geoscience and Remote Sensing (Accepted)*, 2016.

[20] Abdul Nurunnabi, Geoff West, and David Belton, "Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data," *Pattern Recognition*, vol. 48, no. 4, pp. 1404–1419, 2015.

[21] Xiaohu Lu, Jian Yao, Jingge Tu, Kai Li, Li Li, and Yahui Liu, "Pairwise linkage for point cloud segmentation," in *XXIII ISPRS Congress*, 2016.

[22] Yungeun Choe, Seunguk Ahn, and Myung Jin Chung, "Online urban object recognition in point clouds using consecutive point information for urban robotic missions," *Robotics and Autonomous Systems*, vol. 62, no. 8, pp. 1130–1152, 2014.

[23] Yuri Boykov and Gareth Funka-Lea, "Graph cuts and efficient ND image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.