

Article

Automatic Detection and Classification of Pole-Like Objects for Urban Cartography Using Mobile Laser Scanning Data

Celestino Ordóñez ^{1,*}, Carlos Cabo ¹ and Enoc Sanz-Ablanedo ²

¹ Departamento de Explotación de Minas, Grupo de Investigación en Geomática y Computación Gráfica (GEOGRAPH), Universidad de Oviedo, 33004 Oviedo, Spain; carloscabo.uniovi@gmail.com

² Grupo de Investigación en Geomática e Ingeniería Cartográfica (GEOINCA), Universidad de León, Avenida de Astorga, s/n, 24001 Ponferrada, Spain; enocsanz@unileon.es

* Correspondence: ordonezcelestino@uniovi.es; Tel.: +34-985-814-027 or +34-985-458-001

Received: 23 May 2017; Accepted: 19 June 2017; Published: 22 June 2017

Abstract: Mobile laser scanning (MLS) is a modern and powerful technology capable of obtaining massive point clouds of objects in a short period of time. Although this technology is nowadays being widely applied in urban cartography and 3D city modelling, it has some drawbacks that need to be avoided in order to strengthen it. One of the most important shortcomings of MLS data is concerned with the fact that it provides an unstructured dataset whose processing is very time-consuming. Consequently, there is a growing interest in developing algorithms for the automatic extraction of useful information from MLS point clouds. This work is focused on establishing a methodology and developing an algorithm to detect pole-like objects and classify them into several categories using MLS datasets. The developed procedure starts with the discretization of the point cloud by means of a voxelization, in order to simplify and reduce the processing time in the segmentation process. In turn, a heuristic segmentation algorithm was developed to detect pole-like objects in the MLS point cloud. Finally, two supervised classification algorithms, linear discriminant analysis and support vector machines, were used to distinguish between the different types of poles in the point cloud. The predictors are the principal component eigenvalues obtained from the Cartesian coordinates of the laser points, the range of the Z coordinate, and some shape-related indexes. The performance of the method was tested in an urban area with 123 poles of different categories. Very encouraging results were obtained, since the accuracy rate was over 90%.

Keywords: Mobile Laser Scanner (MLS); point cloud; pole-like objects; automatic feature detection; principal component analysis

1. Introduction

Today, efficient and rational management of any sustainable urban environment requires the use of an increasing amount of geospatial information. Mobile laser scanning (MLS) is a relatively new technology that has a promising future as a tool to obtain accurate cartography and construct 3D models, mainly in urban and road environments [1]. The MLS system normally integrates two laser scanners, four to six CCD (charge-coupled device) cameras, one or two GNSS (global navigation satellite system) receivers, an inertial navigation system (INS), a distance measuring indicator (DMI), and software and hardware to collect and process all the data [2]. All these components are placed in a vehicle, so the data are collected while the vehicle is travelling. The main product is a georeferenced 3D point cloud representing the space around the vehicle.

Despite its great capacity to collect data in a short period of time, the fact is that MLS still has some drawbacks that must be gradually corrected to consolidate this technology among cartographers.

One of these drawbacks is the cost of the equipment, which is expected to decrease over time. Another weakness of MLS is concerned with the fact that the raw product is a massive heterogeneous and unstructured cloud point which must be processed to obtain useful information for cartographers. As this process is time-consuming, there is a great interest in developing software devoted to generate cartography automatically. Given that MLS is a developing technology, there is still not much literature concerning the development of algorithms to process MLS point clouds, although algorithms developed for terrestrial laser scanner (TLS) data could also be applied. An overview of different techniques for the extraction of surfaces from point clouds, such as scan line segmentation, surface growing, or clustering methods for the recognition of parameterized surfaces, can be seen in [3,4] proposed a sequential method to classify point clouds in real time by means of a hidden Markov model, although they limit their study to distinguish between vegetation and non-vegetation regions, and between vertical and horizontal surfaces in non-vegetation regions. [5] used a link-chain segmentation algorithm and a classification method to establish different classes of objects in urban environments while they assume that the ground is flat. The classification is based on comparing previously segmented objects with a set of standard, predefined thresholds. Other interesting works related to mobile laser scanning segmentation in urban or road environments, not specifically concerned with pole-like objects, can be seen in [5–12]. The segmentation and classification of urban features can also be performed using data from image-based devices. Segmentation can be implemented directly from the images (e.g., [13,14]), or using dense point clouds derived from them (e.g., [15–17]).

Pole-like objects, such as lampposts, traffic lights, and street trees, are urban elements of interest for cartographers, and consequently some algorithms have been developed for the automatic or semi-automatic detection and/or classification of these objects using MLS point clouds. [18] presented a method for organizing objects (cars, traffic lights, fire hydrants, lampposts, etc.) in 3D point clouds of urban environments obtained by combining MLS and ALS (aerial laser scanning) systems. Once the objects have been detected throughout segmentation, they are classified in different categories using different machine learning techniques such as k-nearest neighbors and support vector machines. The recognition rates are not very high and, moreover, at this moment, the interest is principally in unsupervised methods that do not require operator intervention. [19] detected vertical pole-like objects in a road environment using MLS point clouds by means of a sequential algorithm that includes scan line segmentation, clustering, and classification. The average correctness of the detection was 81.0%. [20] employed a covariance-based procedure to perform a segmentation of MLS point clouds to detect road poles. [21] recognized pole-like objects from MLS point clouds using a Laplacian smoothing followed by principal component analysis for recognizing points on pole-like objects. Pole-like objects with various radii and tilt angles were recognized. [22] developed a method to detect basic structures (trees, traffic signals, lampposts) from MLS data that uses several parameters such as size, shape, orientation, and topological relationships. [23] used voxelization and morphological analysis to detect street trees and some morphological parameters from MLS data. [24] presented an algorithm to detect and classify traffic signs from MLS data. The classification is based on 2D images using a Gaussian-Bernoulli deep Boltzmann machine-based hierarchical classifier. [25] developed a method for the detection and recognition of retro-reflective vertical traffic lights from MLS data. First, the traffic lights are extracted from the point cloud according to geometric and radiometric information. Then, the resulting cluster of points is rasterized and classified in different categories using a linear regression.

In this paper, we develop an algorithm for the automatic detection and classification of five types of pole-like objects from MLS point clouds based on the previous work of [26]. The authors proposed a methodology to detect pole-like street furniture objects based on discretizing the cloud point in voxels and analyzing horizontal sections of the voxel, but although they obtained good results, they did not tackle the problem of classifying the pole-like objects into categories. The distinction between the said elements is carried out using a discriminant analysis algorithm that groups the objects based on the principal component scores obtained from the Cartesian coordinates of the points and on three indexes that are related to the shape of the objects.

2. Materials and Methods

2.1. Data Acquisition

The methodology developed in this paper has been tested on a set of data collected with the Lynx Mobile Mapper system (Teledyne OPtech, Toronto, ON, Canada). The system is based on two LIDAR (light detection and ranging) sensors mounted on a vehicle, an inertial measurement unit (POS LV 520) produced by Applanix (Markham, ON, Canada), which consists of a two-antenna heading measurement system, and an inertial navigation system.

The LIDAR sensors are located in the rear part of the van. Each of them registers points in a plane at 60° to the horizontal and 45° to the longitudinal axis of the vehicle (i.e., the driving direction) with a 360° field of view. The scan frequency of these sensors varies from 80 to 200 Hz, and the pulse repetition rate (PRR) ranges from 75 to 500 kHz. The speed of the van was approximately 40 km per hour. According to the manufacturer, the maximum range is 200 m (20% reflectivity), and the absolute precision of the points registered is ± 5 cm (1σ).

The MLS point cloud was collected in a street in Ourense, a city in the northwest of Spain (Figure 1). It contains 203 million points along 1 km (i.e., on average, 200,000 points per meter along the trajectory of the MLS). The street contains different pole-like objects, particularly trees, lampposts, traffic lights, traffic signals, and traffic panels. The aim of this paper was to detect these objects and separate them into categories.

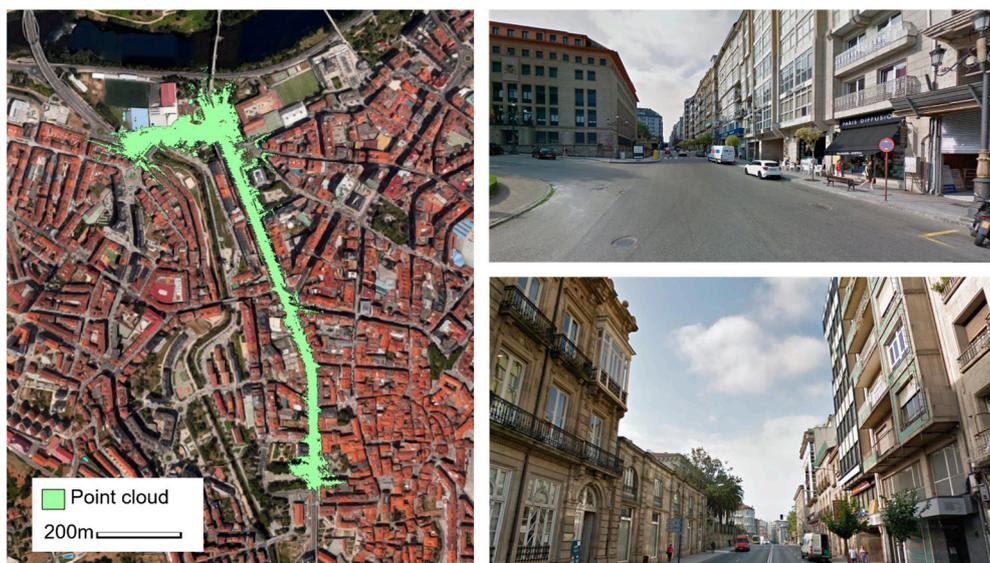


Figure 1. Plan view of the point cloud (left) and two photographs of the street where the point cloud was collected (right).

2.2. Point Cloud Segmentation and Classification

2.2.1. Segmentation

Segmentation of the MLS point cloud was first carried out in order to distinguish pole-like objects from other objects such as buildings, vehicles, or the ground. The segmentation algorithm follows a previous work ([26]). It consists of three consecutive steps: (i) voxelization of the point cloud space; (ii) two-dimensional analysis of horizontal sections of the voxelized point cloud; and (iii) tridimensional reconstruction of the selected features from the previous 2D analysis. In cases where the point cloud is too large for computing the whole dataset at once, the point cloud is automatically divided into smaller subsets with spatial overlap. This division can be performed by splitting the extent of the point cloud into overlapping rectangles, or into stripes along the trajectory of the vehicle.

The different subsets are analyzed separately, and afterwards, duplicate pole-like objects detected in the overlapped areas are joined or eliminated.

Voxelization consists of discretizing the space in a regular tridimensional grid of cubes, and it is used to reduce the amount of data and the processing time. For each of these cubes (voxels) only the coordinates of its centroid and the number of points inside it are stored, given that only this information is needed in the following stages. Moreover, a one-to-one correspondence between voxels and laser points is established.

Once the voxelization has been performed, the tridimensional grid is divided into horizontal slices in order to identify the structures that most likely correspond to the horizontal section of a pole. The identification is carried out by a neighborhood analysis assuming that poles are isolated objects with a small horizontal section. From the centroid of each cell a ring is defined. Poles are searched for by detecting cells inside the ring. The number of points located inside the ring is calculated and a section is rejected when a threshold is surpassed. Figure 2 shows different situations that can take place during the search process.

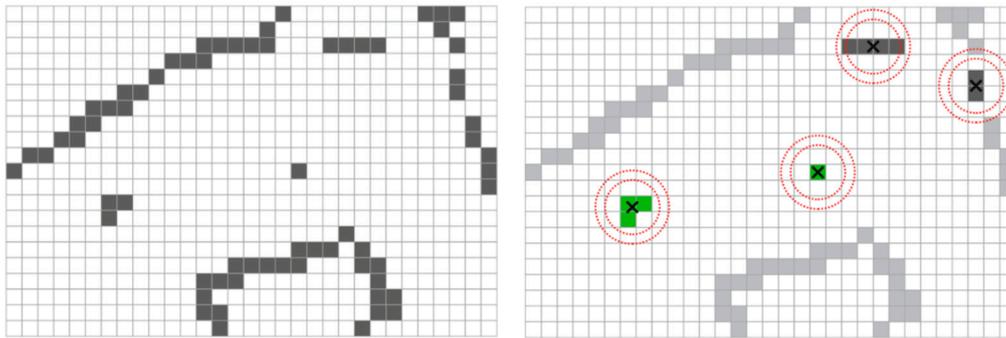


Figure 2. Two-dimensional analysis for pole detection. (Left): Horizontal layer of the voxelized space containing poles and other objects; (Right): Groups of voxels that fulfill all the requirements (green), small groups that do not overcome the isolation filter (dark grey), and large groups of voxels (light grey).

The third stage consists of assembling the cell candidates into poles. All the voxels that share a face, an edge, or a corner are grouped together. Furthermore, vertical continuity and a minimum height are required for these voxel structures to be considered as poles (Figure 3).

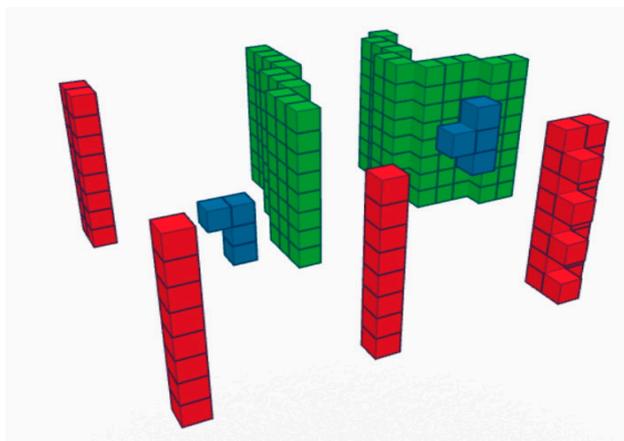


Figure 3. Three-dimensional assembling and analysis. Green groups: voxels in sections that do not fulfill the isolation requirements; Blue groups: voxel groups with valid sections, but without vertical continuity or enough height; Red groups: poles.

2.2.2. Classification

Methods

Once the poles have been detected, we are interested in classifying them into six categories present in the scene: Trees (1), lampposts (2), advertising signals (3), small traffic lights (4), large traffic lights (5), and traffic signs (6). In order to do this, we first extract each of the objects and then apply a supervised classification algorithm. Each of the sample objects was extracted from the point cloud by first detecting the actual pole, as explained in the previous section, and then detecting the top part. All the voxels connected to the top section of each pole are grouped and labeled. Then, using the reversible nature of the voxelization algorithm [26], the points inside the labeled voxels are extracted (Figure 4). These points inherit the labels from the voxels that contain them, and conform the sample objects.

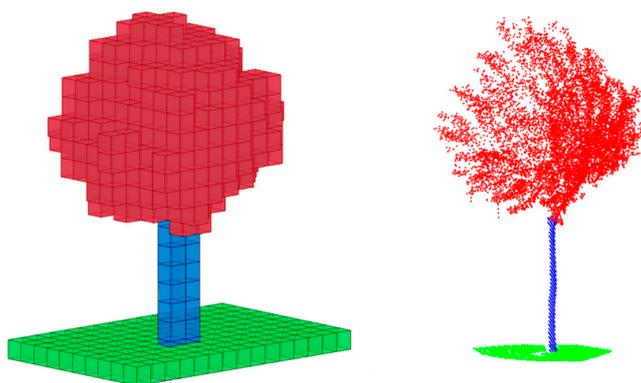


Figure 4. Pole object extraction. (Left): group of voxels defining the actual pole (blue), linked voxels conforming the object attached to the pole (red), and voxels containing points on the ground (green); (Right): points inside each voxel group that inherit the voxel labels.

The classification into different categories was carried out using the well-known linear discriminant analysis (LDA) ([27–29]) and support vector machine (SVM) ([30–33]) techniques for multiple classes, which have been largely applied in many different disciplines. There are many other classification algorithms, but we opted for LDA because of its simplicity, and SVM, for its capacity to determine non-linear boundaries between classes.

LDA is a linear transformation technique that computes the directions representing the axes that maximize the separation between multiple classes. Given a set of observations (x_1, x_2, \dots, x_n) ; $x_i \in \mathbb{R}^d$, d being the number of variables, each of them belonging to a class y_c , $c = 1, \dots, C$, the aim of the LDA is to find, for a new observation x_i , a good predictor y_c of the class it belongs to. Assuming that each of the C classes has a mean μ_j , the within-class and between-class scatter matrices, S_w and S_b , respectively, are

$$S_w = \sum_{j=1}^C \sum_{i=1}^{N_j} (x_{ij} - \mu_j)(x_{ij} - \mu_j)^T; S_b = \frac{1}{C} \sum_{j=1}^C (\mu_j - \mu)(\mu_j - \mu)^T$$

where μ is the mean of the class means, x_{ij} is the i th sample of class j , and N_j is the number of samples in class j . The goal is to maximize the between-class measure while minimizing the within-class measure. One way to do this is by maximizing the ratio of the scatter matrices determinants $R = \frac{|S_b|}{|S_w|}$.

If S_w is a non-singular matrix, this can be accomplished by calculating the eigenvectors of $S_w^{-1}S_b$ corresponding to the largest eigenvalues. These eigenvectors represent the directions of maximum separation between classes.

The multiclass SVM classifier is an extension of the original binary classifier ($y = \{-1, 1\}$) which looks for the maximum-margin hyperplane, that is, the hyperplane representing the largest separation

between the two classes. The algorithm is based on mapping the original finite-dimensional space into a much higher-dimensional space where the boundaries between classes are linear. A hyperplane can be written as the set of observations satisfying $\mathbf{w}\mathbf{x} + b = 0$, where \mathbf{w} represents the normal vector to the hyperplane and b represents the offset. Binary SVM tries to find two parallel hyperplanes equidistant from the maximum-margin hyperplane which separates both classes while the distance between them is as large as possible. Analytically, the solution is formulated as the following minimization problem with restrictions:

$$\begin{cases} \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \\ y_i(\mathbf{w}\mathbf{x}_i + b) \geq 1, i \in \{1, \dots, n\} \end{cases}$$

The solution is a linear combination of a subset of observations near the margin hyperplanes denominated support vectors. Those observations \mathbf{x}_i located near the margin hyperplanes completely determine them, and for that reason they are called support vectors (s.v.):

$$\mathbf{w} = \sum_{s.v.} \beta_i \mathbf{x}_i$$

In this work, we used a mathematical approach based on kernels to solve the SVM model. In particular, a radial basis function (RBF) kernel was used. Given two samples $(\mathbf{x}_i, \mathbf{x}_j)$, the RBF kernel is $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$, $\gamma = \frac{1}{2\sigma^2}$ being a parameter that measures the radius of influence of samples selected by the model as support vectors.

Apart from γ , there is another parameter in SVM that needs to be considered in order to obtain an optimal solution, and it is usually referred as C . C is a generalization parameter that allows the modification of the number of samples selected as support vectors. It controls the tradeoff between smooth decision boundary and classifying the training points correctly. When C is too large, errors are highly penalized in the classification, but the model may overfit. On the other hand, a too small C value may cause underfitting.

Predictor Variables

Except for one variable, the rest of the predictor variables used in the classification are related to the eigenvalues obtained from a principal component analysis (PCA) applied to the coordinates of the laser points. For each of the pole-objects in the sample, the eigenvalues were calculated through an eigendecomposition of the covariance matrix Σ of the laser points $\mathbf{p}_i = (X_i, Y_i, Z_i)$:

$$\Sigma = \sum_{i=1}^N (\mathbf{p}_i - \bar{\mathbf{p}})^T (\mathbf{p}_i - \bar{\mathbf{p}})$$

where N is the number of laser points of the object.

After testing several variables related to the geometric characteristics of the objects under study, finally only four $x_j; j = 1, \dots, 4$, those corresponding to the simplest model of high accuracy, were selected (the discarded covariates tested were λ_1, λ_2 , and λ_3 , the range of λ_1, λ_2 , and λ_3 , and the volume of the convex hull). They were specifically the following:

$x_1 =$ range of Z coordinate. Discriminates between small (traffic signals) and large objects (lampposts, advertising signals).

$x_2 = \frac{\lambda_3}{\lambda_1 \lambda_2}$. Discriminates flat elements (traffic signals) from other objects.

$x_3 = \frac{\lambda_2}{\lambda_3}$. Distinguishes narrow objects, such as traffic signals, from wider elements with similar values of x_2 , such as lampposts.

$x_4 = \frac{\lambda_1 \lambda_3}{\lambda_2^2}$. This variable discriminates between volumetric objects (trees) and flatter objects (traffic signals or lampposts).

where $\lambda_i, i = 1, \dots, 3$, are the three eigenvalues obtained from a PCA (principal component analysis).

Figure 5 shows a sample of the different kinds of pole-like objects extracted from the MLS point cloud. The base is blue and the top part is red. The eigenvectors are also represented in order to show their relationship with the geometry of the objects.

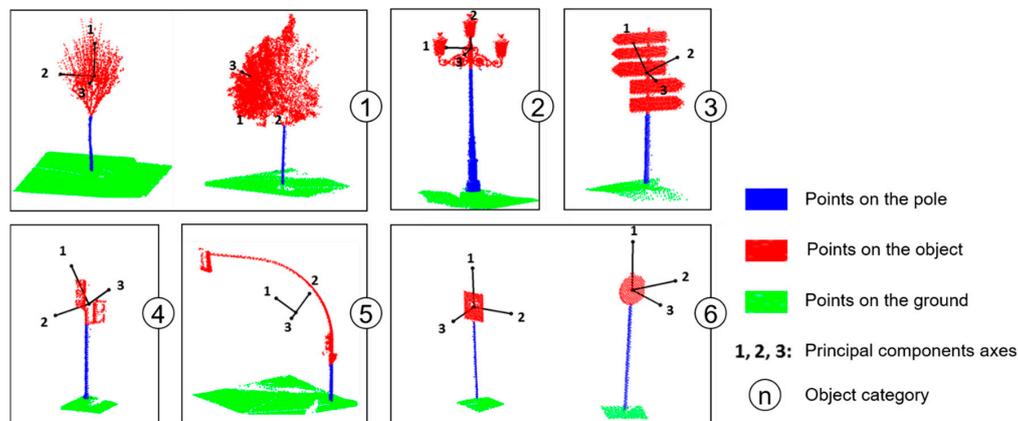


Figure 5. Pole-like object categories extracted from the point cloud.

As can be appreciated in Figure 5, the planar traffic signal has two eigenvalues, corresponding to the eigenvector contained in the plane, which are greater than the third one, which corresponds to the eigenvector normal to that plane. Accordingly, we expect to obtain small values of x_2 for these objects. Lampposts are in some way similar to traffic signs, but they are wider, and then they have a larger value of the variable x_3 . For trees, especially when in leaf, the three eigenvalues tend to have similar values, so we expect to obtain larger values of variable x_3 than for the rest of the objects. As was mentioned, variable x_1 was mainly defined to distinguish between small and large objects.

3. Results

The segmentation and classification methods previously explained were applied to the point cloud described in Section 2.2. All the methods were implemented in Matlab, and the analysis was performed on a commercial laptop (N-551-JB: Core i7, 2.6 GHz, 8 GB DDR3, Asus, Taipei, Taiwan). The point cloud was automatically divided into four overlapping stripes (i.e., each stripe contains points registered by the sensors from a 265-m segment along the trajectory, and the overlap between consecutive stripes is 20 m). The segmentation algorithm was applied individually to the four stripes using a voxel size of 10 cm, and then duplicated pole-like objects were removed. Processing time for the segmentation was 486 s, and 0.2 s for the classification (once the model parameters were determined). The segmentation algorithm detected 91% of the total number of poles in the point cloud, and only 3% of the objects were detected as poles although they were not. Most of these misdetections were almost-vertical branches, so the error could have been avoided by increasing the outer diameter in the segmentation process. As a result, we obtained a sample data with 153 pole-like objects. As this was an unbalanced sample, since there were a greater proportion of trees than other types of objects, we finally removed 30 trees. Accordingly, we obtained a sample of 123 objects with almost the same proportion of elements of each category.

Table 1 shows the confusion matrix obtained after applying LDA and SVM to a sample test formed by 40% (49 objects) of the sample data (the remaining 74 objects were used to train the algorithms). In order to select the optimal values for parameters γ and C of the SVM classifier, a cross-validation procedure was applied to the training dataset for each pair of values $(\gamma_j, C_j), j = 1, \dots, m$ on a grid in the interval $[2^{-10}, 2^{10}]$. Columns refer to existing pole-types and rows refer to predicted pole-types. The last column represents errors of commission (EC, or the ratios of poles classified as a type that is different from the true type) and the last row represents errors of omission (EO, or the ratio of poles representing a specific type that have been classified in another category). The last two rows register

the overall accuracy (ACC) and the kappa index (k). ACC is the fraction of correctly classified poles with regard to all the poles in the sample. k is also a measure of the accuracy in the classification, but it takes into account the possibility of the agreement occurring by chance:

$$k = \frac{\text{observed accuracy} - \text{chance agreement}}{1 - \text{chance agreement}}$$

Table 1. Confusion matrix for the LDA and SVM models.

	LDA							EC	SVM							EC
	1	2	3	4	5	6	1		2	3	4	5	6			
1	10	0	0	1	1	0	0.17	1	11	0	0	1	0	0	0.08	
2	0	8	0	0	0	0	0.00	2	0	8	0	0	0	0	0.00	
3	0	0	3	0	0	0	0.00	3	0	0	3	0	0	0	0.00	
4	0	0	0	10	0	0	0.00	4	0	0	0	10	0	0	0.00	
5	0	0	1	0	4	0	0.20	5	0	0	0	0	5	0	0.00	
6	0	0	0	0	0	11	0.00	6	0	0	1	0	0	10	0.09	
EO	0.00	0.00	0.25	0.09	0.20	0.00	0.06	EO	0.00	0.00	0.25	0.09	0.00	0.00	0.04	
ACC	0.94							ACC	0.96							
k	0.93							k	0.95							

Thus, the results obtained confirm that the four variable predictors used to classify the objects are appropriate, given the high accuracy reached with both classification models.

After repeating the analysis with different training-test samples, it was found that class 3 (advertising panels) was, in general, the worst classified. In fact, we detect a trend whereby some advertising panels are misclassified as lampposts or traffic signs.

Figure 6A shows the distribution of 28 poles and the objects attached to them in a section of the test street. Figure 6B presents the results of the SVM classification on these objects, including an advertisement panel classified as a traffic sign. This misclassification could be due to the reduced size of this specific panel, whose size and shape are similar to some rectangular traffic signs.

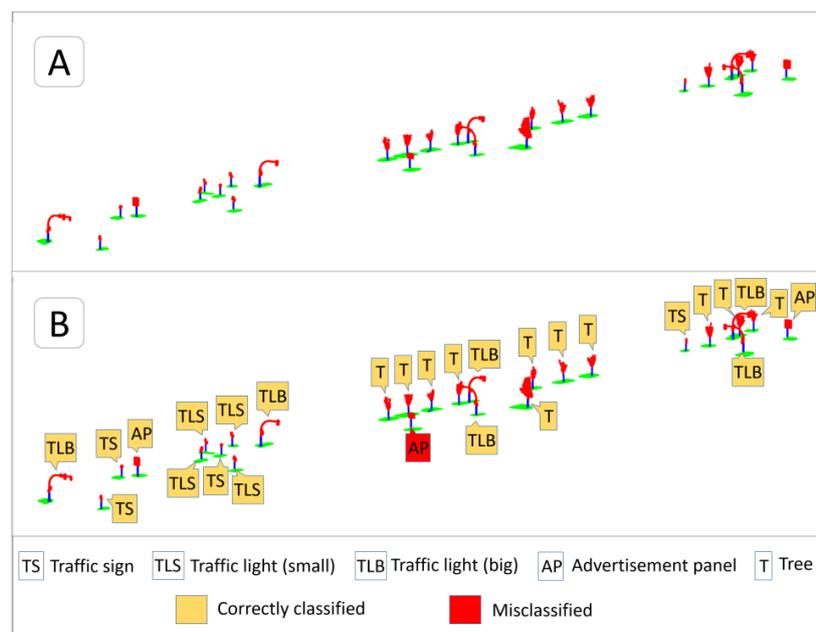


Figure 6. Pole-like object segmentation and classification in a stretch of the test street. (A) Poles, features attached to them, and a portion of ground around each pole following the same color schema as in Figure 5; (B) Classification and labeling of the objects on segmented poles using SVM.

4. Conclusions

A methodology for the recognition of different types of pole-like objects from mobile laser scanning point clouds was developed. First, we detect the objects using a heuristic segmentation algorithm that assumes that poles are isolated and almost vertical objects. Secondly, linear discriminant analysis (LDA) and support vector machine (SVM) algorithms were used to distinguish between six types of pole-like objects.

After trying several shape-related predictor variables to carry out the classification of the objects, we finally select only four of them, consisting of those that provide the best prediction with the minimum number of variables. One of those variables is the range of the Z coordinate and the other three are simple algebraic expressions that relate the eigenvalues of a principal component analysis (PCA) conducted on the Cartesian coordinates.

The application of our methodology to a dataset produced very good results for both models (LDA and SVM) with very low error rates and high kappa indices. Errors of commission and omission were also very low, with the class of advertising panels providing the worst results.

Acknowledgments: This paper has been funded by Project FC-15-GRUPIN14-033 of the Fundación para el Fomento en Asturias de la Investigación Científica Aplicada y la Tecnología (FEDER support included).

Author Contributions: Celestino Ordóñez proposed the research and carried out the classification analysis. Carlos Cabo developed the segmentation algorithm and defined and calculated the predictor variables. Enoc Sanz contributed to obtain the database, analyze and discuss the results and supervised the whole article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nebiker, S.; Bleisch, S.; Christen, M. Rich point clouds in virtual globes—A new paradigm in city modeling? *Comput. Environ. Urban. Syst.* **2010**, *34*, 508–517. [[CrossRef](#)]
2. Puente, I.; González-Jorge, H.; Arias, P.; Armesto, J. Land-based Mobile Laser Scanning Systems: A review. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2012**, *38*, 163–168. [[CrossRef](#)]
3. Vosselman, G.; Gorte, B.G.; Sithole, G.; Rabbani, T. Recognising structure in laser scanner point clouds. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2004**, *46*, 33–38.
4. Hadjiliadis, O.; Stamos, I. Sequential Classification in Point Clouds of Urban Scenes. In Proceedings of the 5th International Symposium 3D Data Processing, Visualization and Transmission, Paris, France, 17–20 May 2010.
5. Aijazi, A.K.; Checchin, P.; Trassoudaine, L. Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation. *Remote Sens.* **2013**, *5*, 1624–1650. [[CrossRef](#)]
6. Brenner, C. Extraction of Features from Mobile Laser Scanning Data for Future Driver Assistance Systems. In *Advances in GIScience*; Sester, M., Bernard, L., Paelke, V., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 25–42.
7. Rutzinger, M.; Höfle, B.; Oude Elberink, S.; Vosselman, G. Feasibility of façade footprint extraction from Mobile Laser Scanning Data. *Photogramm. Fernerkund. Geoinform.* **2011**, 97–107. [[CrossRef](#)]
8. Jochem, A.; Hofle, B.; Rutzinger, M. Extraction of Vertical Walls from Mobile Laser Scanning Data for Solar Potential Assessment. *Remote Sens.* **2011**, *3*, 650–667. [[CrossRef](#)]
9. El-Halawany, S.; Moussa, A.; Lichti, D.D.; El-Sheimy, N. Detection of road curb from mobile terrestrial laser scanner. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science, Calgary, AB, Canada, 29–31 August 2011.
10. Heo, J.; Jeonga, S.; Parka, H.; Junga, J.; Hanb, S.; Honga, S.; Sohna, H. Productive high-complexity 3D city modeling with point clouds collected from terrestrial LiDAR. *Comput. Environ. Urban. Syst.* **2013**, *41*, 26–38. [[CrossRef](#)]
11. Boulaassal, H.; Landes, T.; Grussenmeyer, P.; Tarsha-Kurdi, F. Automatic segmentation of building facades using terrestrial laser data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2007**, 65–70.
12. Buch, N.; Orwell, J.; Velastin, S.A. Detection and classification of vehicles for urban traffic scenes. In Proceedings of the Visual Information Engineering 5th International Conference, Xi'an China, 29 July–1 August 2008; pp. 182–187.
13. Balali, V.; Golparvar-Fard, M. Segmentation and Recognition of Roadway Assets from Car-Mounted Video Streams using Scalable Non-Parametric Image Parsing. *Elsevier J. Autom. Constr.* **2015**, *49*, 27–39. [[CrossRef](#)]

14. Tighe, J.; Niethammer, M.; Lazebnik, S. Scene parsing with object instances and occlusion ordering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2014, Columbus Convention Center Columbus, OH, USA, 23–28 June 2014; pp. 3748–3755.
15. Balali, V.; Jahangiri, A.; Machiani, S.G. Multi-class US traffic signs 3D recognition and localization via image-based point cloud model using color candidate extraction and texture-based recognition. *J. Adv. Eng. Inform.* **2017**, *32*, 263–274. [[CrossRef](#)]
16. Brostow, G.; Shotton, J.; Fauqueur, J.; Cipolla, R. Segmentation and recognition using structure from motion point clouds. *Comput. Vis.–ECCV* **2008**, 44–57. [[CrossRef](#)]
17. Gevaert, C.M.; Persello, C.; Sliuzas, R.; Vosselman, G. Informal settlement classification using point-cloud and image-based features from UAV data. *ISPRS J. Photogramm. Remote Sens.* **2017**, *125*, 225–236. [[CrossRef](#)]
18. Golovinskiy, A.; Kim, V.G.; Funkhouser, T. Shape-based recognition of 3d point clouds in urban environments. In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 2154–2161.
19. Lehtomäki, M.; Jaakkola, A.; Hyyppä, J.; Kukko, A.; Kaartinen, H. Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data. *Remote Sens.* **2010**, *2*, 641–664. [[CrossRef](#)]
20. El-Halawany, S.; Lichti, D. Detection of road poles from mobile terrestrial laser scanner point cloud. In Proceedings of the 2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM), Xiamen, China, 10–12 January 2011.
21. Yokoyama, H.; Date, H.; Kanai, S.; Takeda, H. Pole-like objects recognition from Mobile Laser Scanning Data using smoothing and principal component analysis. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2011**, *38*, 115–120. [[CrossRef](#)]
22. Pu, S.; Rutzinger, M.; Vosselmand, G.; Elberink, S.O. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, S28–S39. [[CrossRef](#)]
23. Wu, B.; Yu, B.; Yue, W.; Shu, S.; Tan, W.; Hu, C.; Huang, Y.; Wu, J.; Liu, H. A Voxel-Based Method for Automated Identification and Morphological Parameters Estimation of Individual Street Trees from Mobile Laser Scanning Data. *Remote Sens.* **2013**, *5*, 584–611. [[CrossRef](#)]
24. Yu, Y.; Li, J.; Wen, C.; Guan, H.; Luo, H.; Wang, C. Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data. *ISPRS J. Photogramm. Remote Sens.* **2016**, *113*, 106–123. [[CrossRef](#)]
25. Riveiro, B.; Díaz-Vilariño, L.; Conde-Carnero, B.; Soilán, M.; Arias, P. Automatic Segmentation and Shape-Based Classification of Retro-Reflective Traffic Signs from Mobile LiDAR Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *9*, 295–303. [[CrossRef](#)]
26. Cabo, C.; Ordoñez, C.; García-Cortés, S.; Martínez, J. An algorithm for automatic detection of pole-like street furniture objects from Mobile Laser Scanner point clouds. *ISPRS J. Photogramm. Remote Sens.* **2014**, *87*, 47–56. [[CrossRef](#)]
27. Fisher, R.A. The Use of Multiple Measurements in Taxonomic Problems. *Ann. Eugen.* **1936**, *7*, 179–188. [[CrossRef](#)]
28. Rao, R.C. The utilization of multiple measurements in problems of biological classification. *J. R. Stat. Soc. Ser. B* **1948**, *10*, 159–203.
29. McLachlan, G.J. *Discriminant Analysis and Statistical Pattern Recognition*; Wiley Interscience: Hoboken, NJ, USA, 2004.
30. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1995; p. 188.
31. Lodha, S.; Kreps, E.; Helmbold, D.; Fitzpatrick, D. Aerial LiDAR Data Classification Using Support Vector Machines (SVM). In Proceedings of the Third International Symposium on 3D Processing, Visualization and Transmission, Chapel Hill, NC, USA, 14–16 June 2006; IEEE Xplore Digital Library: New York, NY, USA, 2006.
32. Abe, S. *Support Vector Machines for Pattern Classification*; Springer: London, UK, 2010; p. 471.
33. Yang, B.; Dong, Z. A shape-based segmentation method for mobile laser scanning point clouds. *ISPRS J. Photogramm. Remote Sens.* **2013**, *81*, 19–30. [[CrossRef](#)]

