Detection of utility poles from noisy Point Cloud Data in Urban environments

Alex Ferrin, Julio Larrea, Miguel Realpe, Daniel Ochoa
Escuela Superior Politécnica del Litoral, ESPOL, FIEC
Campus Gustavo Galindo Km 30.5 Vía Perimetral, Guayaquil, Ecuador
[aferrin, juallarr, mrealpe, dochoa]@espol.edu.ec

ABSTRACT
In recent years 3D urban maps have become more common, thus providing complex point clouds that include diverse urban furniture such as pole-like objects. Utility poles detection in urban environment is of particular interest for electric utility companies in order to maintain an updated inventory for better planning and management. The present study develops an automatic method for the detection of utility poles from noisy point cloud data of Guayaquil - Ecuador, where many poles are located very close to buildings, which increases the difficulty of discriminating poles, walls, columns, fences and building corners. The proposed method applies a segmentation stage based on clustering with vertical voxels and a classification stage based on neural networks.

CCS Concepts
Computing methodologies → Artificial intelligence
Computing methodologies → Computer graphics

Keywords
Pole Detection; 3D Classification; Point Cloud; 3D Recognition; Neural Network;

1. INTRODUCTION
Detection and classification of pole-like objects in urban environments is a topic that has been analyzed and studied for various purposes: Advanced Driver Assistant System [1], development of precise mappings [2], city modeling and management [3], and for various applications in those kind of areas. For city management, detailed environment information is needed in order to plan land use and make better decision about urban furniture. Utility poles are of particular interest for electric utility companies, since they need to keep an updated inventory of overhead power lines and various other public utilities, such as electrical cable, fiber optic cable, and related equipment such as transformers and street lights.

The Ecuadorian state power distribution holding company "Corporación Nacional de Electricidad - CNEL" leases the use of utility poles to Internet and TV providers. However, the lack of an updated inventory makes difficult to keep a real tracking of the quantity of poles used by those companies. In order to solve this problem, CNEL has decided to digitize some cities of Ecuador using a vehicle-based laser scanning (VLS). Currently, a manual process is being used in order to detect and tag the utility poles in the digital map. Nevertheless, it is a slow, tedious and expensive task. The objective of the present work is to handle point cloud data provided by CNEL in order to automatically create an inventory of the utility poles in a city, or to provide information to an assistant system to reduce the manual process.

AICCC '18, December 21–23, 2018, Tokyo, Japan

ACM ISBN 978-1-4503-6623-6/18/12...$15.00
DOI: https://doi.org/10.1145/3299819.3299829

2. RELATED WORK
The classification and detection of poles in urban areas has been approached using different methods on different investigations: In [2], vertical object detection is done through an anomaly detection algorithm RX and classified through a clustering algorithm. [4] uses neural networks to classify based on the normalized cell values on each image, and in [5] Linear Discriminant Analysis and Support Vector Machines are used for classification of the distinct Pole-Like objects, using as predictor variables simple algebraic expressions with PCA analysis of the point clouds.

Since pole-Like objects have a cylindrical shape, many works are based on detecting cylinders [6, 7]. A common approach is also based on finding circle arcs in horizontal cross-section of the point clouds [8-11]. In [10] individual scan lines are processed for detecting arcs, then clusters are created by adding arcs either below the current cluster or above. However the method cannot detect poles with a specific radio. A circular model with an adaptive radius is proposed in [11] in order to find arcs of different sizes, and then a vertical region growing algorithm is used for clustering the points into the pole-like objects.

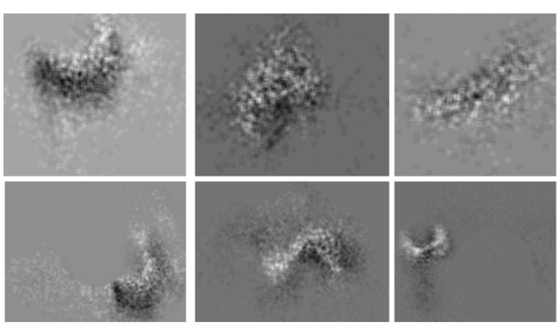
In [12] the data from the surrounding of the pole-like objects is used to create context features in order to classify objects into three classes: utility poles, lamp posts, and street signs.

3. PROPOSED APPROACH
3.1 Problem definition
A common pre-processing stage for segmenting pole-like elements consist of removing large vertical surfaces in order to eliminate columns of buildings, fences and building corners, that are not objects of interest and may be confused with poles. However, in some Latin-American countries, such as Ecuador, many electrical poles are located next to buildings, or houses are built until the border of the sidewalk getting very close to poles (Figure 1). Thus, in the environment of this work is difficult to avoid these false positives unless some poles are lost. Consequently, the segmentation approach in this research does not remove large vertical surfaces, and breaks them in small units instead. Additionally, some parts of the point cloud provided by CNEL include calibration errors that deform the cylindrical shape

of poles (Figure 2). Thus, the horizontal cross-section layers approach was discarded because the 2D projection of poles was very irregular (Figure 3). This paper is proposing two stages: Segmentation and Classification. The segmentation preprocesses the cloud point in order to generate individual objects that may represent utility poles, while the classification creates a model to discriminate poles from other objects.

Figure 1. Utility poles close to buildings (houses).

Figue 2. Calibration errors present in the point cloud.



Figue 3. 2D projection of poles

3.2 Segmentation process

Since the ground joins all the objects, the first step of the segmentation process is to eliminate points corresponding to the ground and then groups the remaining points according to their distances using the basic method for clustering and segmentation presented in [13]. For this, the method described in [14] was follow. First, a RANSAC model [15] is employed to find the largest planar horizontal surface in order to create a model of the ground. Then, all the points inside a threshold of 15 cm from the plane are discarded, as shown in Figures 4 and 5.

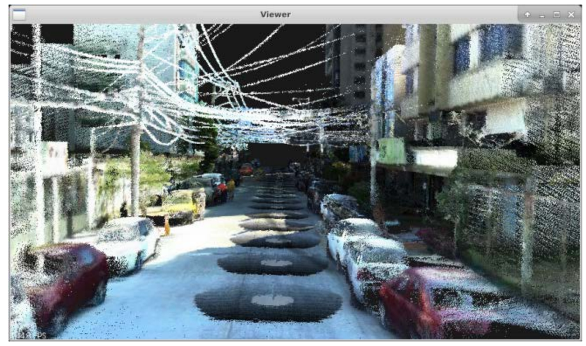


Figure 4. Point Cloud of a street in Guayaquil

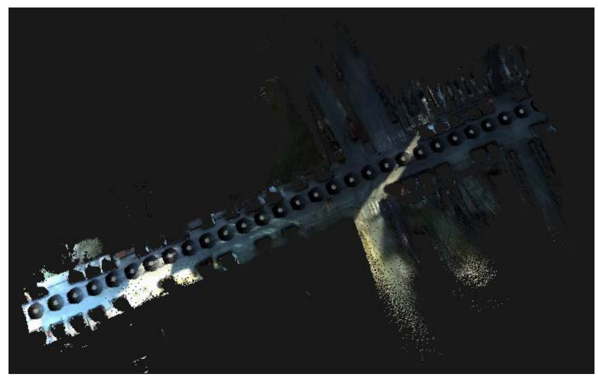


Figure 5. Ground model of a street in Guayaquil

Since some utility poles in Ecuador are built very close to buildings (touching them), it is necessary to split the cloud point in order to obtain the independent vertical objects. For this, a voxel filter [16] was applied in order to divide the point cloud space in a 3D grid of vertical voxels whose horizontal cross-section resolution is similar to the horizontal cross-section of a pole. Finally, the points are clustered with their nearest neighbours based on the Euclidean distance metric following the steps presented in [13]. The result of the segmentation process is shown in Figure 6.

Figure 6. Segmented point clouds

3.3 Classification process

With the segmentation process finished, a general geometric rule (width, height) is applied to remove the point clouds that belong to objects that are clearly not utility poles. Consequently, point clouds of individual objects with geometric characteristics similar to a pole are obtained (Figure 7). The next step is the classification, this step is done by first extracting the features from the point clouds, and later passing those as the input vector of a neural network, which will then classify the objects into their corresponding classes.

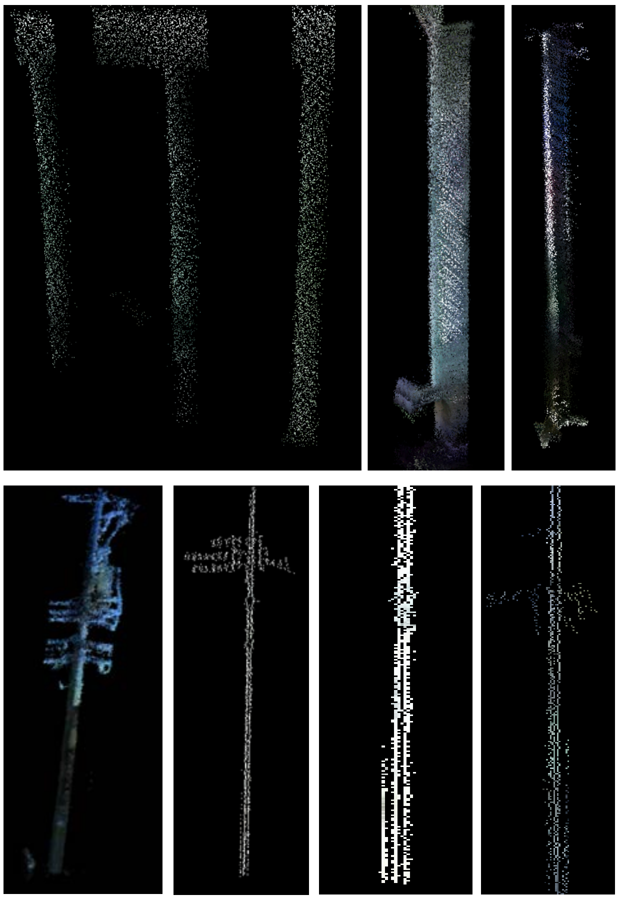


Figure 7. Point cloud of top) columns and part of buildings. bottom) utility poles.

3.3.1 Preprocessing stage

Before entering the classification stage, preprocessing must be done on the data, to keep it homogeneous and extracting the features of the objects to be classified. A C++ program was made using the PCL library to work with the point clouds of the individual objects, filtering the top part of the objects and extracting the features. According to [5], some feature can be extracted using the eigenvalues of the PCA analysis; such variables are:

– X_1: Height of the object

– $X_2 = \lambda3/(\lambda1 * \lambda2)$: Differentiates flat objects from other objects

– $X_3 = \lambda2/\lambda3$: Differentiates narrow objects and wider ones with similar values of X_2

– $X_4 = (\lambda1 * \lambda3)/(\lambda2)^2$: Differentiates volumetric objects, such as tree, from others.

Where λ1, λ2, λ3, are the eigenvalues obtained from the PCA analysis.

Aside from those variables, plane objects can be further differentiated by calculating the variance of the surface normals on the three axes, as most surface normals on plane objects tends to have the same direction, thus having a small variance, in contrast to other elements whose surface normals point to various directions. So, 3 more variables were added to as features:

– X_5: VarianceNX (The variance of the normals in X axis)

– X_6: VarianceNY (The variance of the normals in Y axis)

– X_7: VarianceNZ (The variance of the normals in Z axis)

In total, 7 variables were extracted as features (Table 1) for the classification of the pole-like objects, and such were saved in a CSV file, for further processing on the next step.

Table 1. Example of features extracted from point cloud

X_1	X_2	X_3	X_4	X_5	X_6	X_7
11988	2.2E-9	128.86	0.024	0.017	0.005	0.013
11997	6.5E-8	3.638	4.337	0.346	0.257	0.066
11970	5.3 E-8	4.689	9.317	0.315	0.283	0.024
7015	6.2 E-7	2.218	1.088	0.262	0.231	0.172
10389	1.8 E-8	14.209	0.558	0.076	0.04	0.042
11583	1.5 E-8	14.44	1.164	0.202	0.164	0.093
6608	9.7 E-7	2.466	0.843	0.223	0.167	0.225
6933	2.5 E-7	1.5	3.85	0.264	0.231	0.213
11641	1.3 E-8	18.529	0.497	0.153	0.12	0.142

3.3.2 Classification stage

Once the features were extracted, the classification stage was performed. Classification is done using a Feed-Forward Neural network, to classify the objects in 5 distinct classes: trees, palms, poles, flat objects and other objects.

The neural network was designed on MATLAB (R2017b version) using the Neural Network Toolbox it provides.

The neural network structure is as follows:

- 7 neurons in the input layer
- 5 neurons in the output layer
- 12 neurons in the hidden layer
- Sigmoid function on the neurons in the hidden layer
- Softmax function on the neurons in the output layer
- Bayesian Regularization as the training function

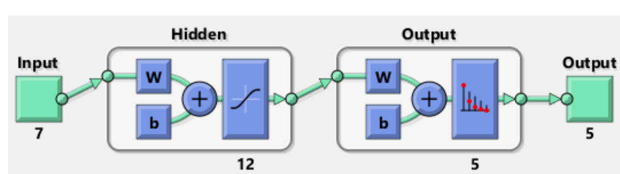


Figure 8. Neural network design diagram.

4. EXPERIMENTAL RESULTS

The proposed approach was tested in a sample from the 3D map of Guayaquil. The sampled cloud contained around 800 million points and represented an urban area that included some parks, buildings and houses. In this section, the results of the approach presented in this work are shown. The results of the segmentation step are 814 point clouds representing 5 classes of objects, illustrated in Figure 9.

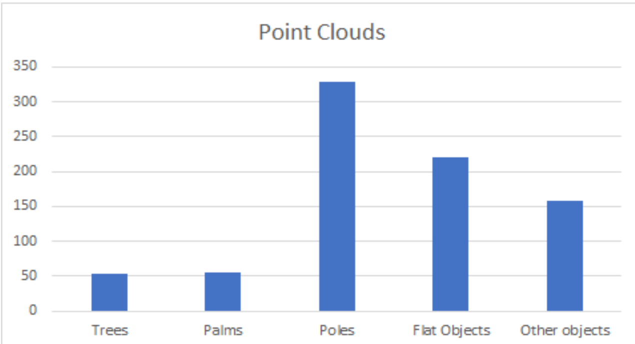


Figure 9. Distribution of objects.

The data then was passed to the preprocessing module, where, by applying the PCA analysis and estimating the surface normals, the features needed were extracted from the point clouds.

Those objects were divided into 2 proportionate sets. One set with 70% of the data of each class, the Training Set, and the other set with the remaining 30% of the data, the Testing Set.

The Training Set was used to train the network, varying the number of neurons in the hidden layer, to find the best settings for the network. The best setting was found using 12 neurons in the hidden layer, which results are shown in the confusion matrix in Figure 10.

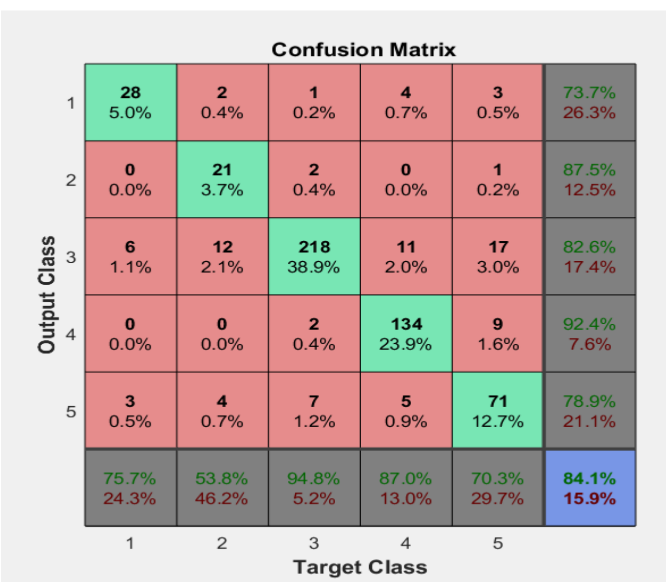


Figure 10. Training confusion matrix

Later, the Testing Set was used to check the ability of the neural network to generalize the data, obtaining the confusion matrix in Figure 11.

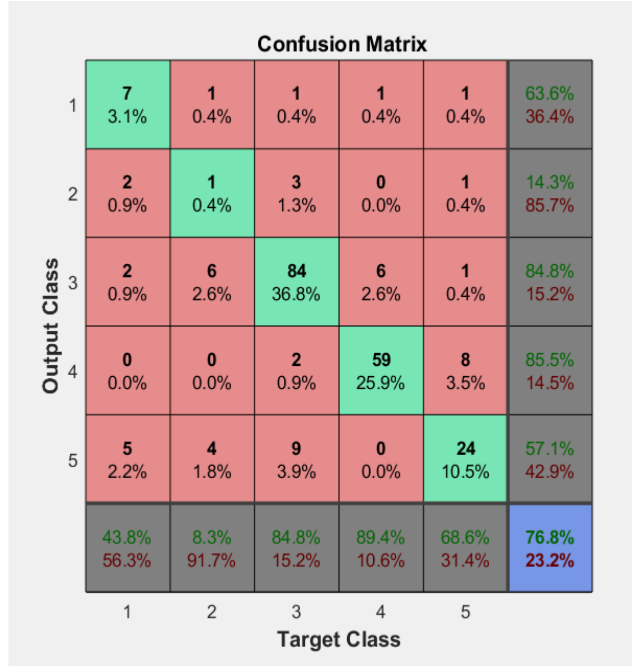


Figure 11: Testing confusion matrix

As it can be seen, the classification percentage is around 85%, having the higher error on elements like trees and palms, followed by other objects. Too many things can be said about this, and those will be discussed in the next section.

5. CONCLUSIONS AND FUTURE WORKS

5.1 Conclusions

This work can be concluded with the following findings.

- The sample size of trees and palms is low in comparison to the rest of the samples, as such; the neural network can't make a robust model of those, and tends to make mistakes in those classes.
- The resultant point clouds obtained from the segmentation process were too noisy, and further processing would be required in order to effectively differentiate the objects.
- Segmenting objects to focus on specific areas is beneficial to calculate efficiently the variables without much margin of error. That process cut off noisy data such as persons, plants, objects that can be in the low part of the pole.
- Even though the proposed approach do not automatically detects all the utility poles present in the noisy point cloud data, it can be used to provide information to an assistant system in order to reduce the manual process which is slow, tedious and expensive.

5.2 Future Works

As it was concluded, there are some factors that limit the algorithm to get a higher success rate. So, in fact, there can be some future works that can be done to improve the success rate of classification.

- Segmentation is a vital part when we talk about searching patterns or classification problems, eliminating noise as much as possible must be done in the segmentation process, and that, as a result, the accuracy of the features calculated and inserted in the neural network will improve, allowing differentiating the objects efficiently.

- When cameras and sensors are used to obtain digital information of real world objects, it is necessary to focus all the possible angles, so in that way we can extract all the information of that object without missing important facts that can be used in classification process.
- Having a homogeneous sample size for each kind of object would be favorable, so the neural network could build a robust model for each output class.
- The use of Convolutional neural network is being analyzed in order to improve the classification results.

6. REFERENCES

[1] Levinson, J.; Askeland, J.; Becker, J.; Dolson, J.; Held, D.; Kammel, S.; Kolter, J.Z.; Langer, D.; Pink, O.; Pratt, V. "Towards fully autonomous driving: Systems and algorithms". *In Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, 5–9 June 2011; pp 163–168.

[2] Borja Rodríguez Cuenca, Silverio García Cortés, Celestino Ordóñez and Maria C. Alonso. "Automatic Detection and Classification of Pole-Like Objects in Urban Point Cloud Data Using an Anomaly Detection Algorithm" 28 September 2015.

[3] Alberto Broggi, Pietro Cerri, Paolo Medici, Pier Paolo Porta. "Real Time Road Signs Recognition" *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium Istanbul*, Turkey, June 13-15, 2007.

[4] D. Habermann, A. Hata, D. Wolf and F. Osorio, "Artificial Neural Nets Object Recognition for 3D Point Clouds", 2013 Brazilian Conference on Intelligent Systems, 2013.

[5] C. Ordóñez, C. Cabo and E. Sanz-Ablanedo, "Automatic Detection and Classification of Pole-Like Objects for Urban Cartography Using Mobile Laser Scanning Data", *Sensors*, vol. 17, no. 7, p. 1465, 2017.

[6] Bienert, A.; Maas, H.-G.; Scheller, S. "Analysis of the Information Content of Terrestrial Laser scanner Point Clouds for the Automatic Determination of Forest Inventory Parameters". *In Proceedings of the International Workshop on 3D Remote Sensing in Forestry*, Vienna, Austria, 2006; pp. 55-60.

[7] Luo, D.; Wang, Y. Rapid "Extracting Pillars by Slicing Point Clouds. Int. Arch. Photogramm". *Remote Sens.* 2008, XXXVII–B3b, 215-218.

[8] Fukano, K.; Masuda, H. "Detection and classification of pole-like objects from mobile mapping data". *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci*. 2015, 2, 57–64.

[9] Y. Mori, K. Kohira, and H. Masuda, "Classification of pole-like objects using point clouds and images captured by mobile mapping systems", *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-2, 731-738, 2018

[10] Lehtomäki, Matti, Anttoni Jaakkola, Juha Hyyppä, Antero Kukko and Harri Kaartinen, "Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data". *Remote Sensing* 2 (2010): 641-664, 2010.

[11] Kang, Zhizhong & Yang, Juntao & Zhong, Ruofei & Wu, Yongxing & Shi, Zhenwei & Lindenbergh, Roderik. "Voxel-Based Extraction and Classification of 3-D Pole-Like Objects From Mobile LiDAR Point Cloud Data". *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 11. 1-12. 10.1109/JSTARS.2018.2869801, 2018.

[12] Yokoyama, H & Date, Hiroaki & Kanai, Satoshi & Takeda, H. "Detection and classification of pole-like objects from mobile laser scanning data of urban environments". *Int. J. CAD/CAM.* 13. 31-40, 2013.

[13] R. Rusu. "Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments", Computer Science department, Technische Universitaet Muenchen, Germany, 2009.

[14] Miguel Realpe1, Boris Vintimilla, Ljubo Vlacic, "Towards Fault Tolerant Perception for autonomous vehicles: Local Fusion", *7th IEEE International Conference on Robotics, Automation and Mechatronics (RAM)*, Angkor Wat – Cambodia, 2015.

[15] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, pp. 381-395, 1981.

[16] Douillard, B.; Underwood, J.; Kuntz, N.; Vlaskine, V.; Quadros, A.; Morton, P.; Frenkel, A. "On the segmentation of 3D LiDAR point clouds". *In Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 9–13 May 2011; pp 2798–2805.