

# Post-print on Author's Personal Website: James C Koch

Article Name with DOI link to post-print published Version complete citation:

- Gargoum, Suliman A. and Koch, James C. and El-Basyouny, Karim. A Voxel-Based Method for Automated Detection and Mapping of Light Poles on Rural Highways Using LiDAR Data. Proceedings of the Transportation Research Board 97th Annual Meeting. Transportation Research Record. <https://doi.org/10.1177/0361198118787657>

Post-print

As per publisher copyright is ©2018




This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.



Article Post-print starts on the next page →

# A Voxel-Based Method for Automated Detection and Mapping of Light Poles on Rural Highways using LiDAR Data

Transportation Research Record  
1–10  
© National Academy of Sciences:  
Transportation Research Board 2018  
Reprints and permissions:  
sagepub.com/journalsPermissions.nav  
DOI: 10.1177/0361198118787657  
journals.sagepub.com/home/trr  


Suliman A. Gargoum<sup>1</sup>, James C. Koch<sup>1</sup>, and Karim El-Basyouny<sup>1</sup>

## Abstract

The number of light poles and their position (in terms of density and offset off the roadside) have significant impacts on the safe operation of highways. In current practice, inventory of such information is performed in periodic site visits, which are tedious and time consuming. This makes inventory and health monitoring of poles at a network level extremely challenging. To relieve the burden associated with manual inventory of poles, this paper proposes a novel algorithm which can automatically obtain such information from remotely sensing data. The proposed algorithm works by first tiling point cloud data collected using light detection and ranging (LiDAR) technology into manageable data tiles of fixed dimensions. The data are voxelized and attributes for each data voxel are calculated to classify them into ground and nonground points. Connected components labeling is then used to perform 3D clustering of the data voxels. Further clustering is performed using a density-based clustering to combine connected components of the same object. The final step involves classifying different objects into poles and non-poles based on a set of decision rules related to the geometric properties of the clusters. The proposed algorithm was tested on a 4 km rural highway segment in Alberta, Canada, which had substantial variation in its vertical alignment. The algorithm was accurate in detecting nonground objects, including poles. Moreover, the results also highlight the importance of considering the length of the highway and its terrain when detecting nonground objects from LiDAR.

Periodic assessment of road infrastructure is essential for safe and efficient operation of roads. Whether it is street-light poles, traffic signs, bridges, or pavement surface, all these assets must be routinely assessed to ensure they meet the necessary standards throughout the road's service life. The density of light poles, their dimensions, and placement have significant impact on the safety of a highway. In fact, the relationship between the presence of light poles on a highway and safety is a complex one. For instance, while increasing the light pole density may help reduce some types of night-time crashes as a result of the improved visibility, the greater density could also result in more fixed object collisions.

Many studies have found a significant relationship between the offset of poles from the road and the safety of highways (1, 2). Owing to the significant damage poles could cause in fixed object collisions, design guides require that they are placed at a certain offset from the roadside, depending on their sizes and dimensions. This helps in the efforts to design more forgiving highways where a driver is given the best chance of recovery in the case of a collision.

Developing an inventory of light poles and their locations on a highway corridor is also essential to increase the efficiency of maintenance and street aesthetics. Unfortunately, in current practice, inventory information is obtained in manual inspections which are error prone and require several man-hours of work. Obtaining such information on a network level is therefore expensive, and constantly updating existing records is often infeasible. In recent years, light detection and ranging (LiDAR) technology has been considered as an alternative for the inventory and assessment of many elements of a road's infrastructure (3–7).

LiDAR technology combines laser scanners, global navigation satellite systems (GNSS) and inertial measurement units (IMU) into one system to collect a 3D point cloud of the road's environment. In mobile laser

<sup>1</sup>Department of Civil and Environmental Engineering, University of Alberta, Edmonton, AB, Canada

## Corresponding Author:

Address correspondence to Suliman A. Gargoum: gargoum@ualberta.ca

scanning (MLS), the scanning system is mounted on a truck that travels down a highway, creating a 360° virtual image of the highway consisting of closely spaced points of known positional coordinates and intensity information. Although not useful for pole extraction because of the low-reflective surface texture, intensity information is commonly used to extract highly reflective objects such as traffic signs and lane markings. The high speeds at which data are collected enables efficient and safe mapping of extensive areas.

Although a fair number of studies have attempted the extraction of poles from LiDAR, the majority of those algorithms have been developed for short, flat segments in an urban setting. Consequently, there is a lack of studies in which pole detection is attempted on high-speed (>80 km/h) rural highway with rolling terrain. Unlike flat urban roads, rural highways often experience significant variation in vertical alignments. This creates huge challenges when extracting the nonground surface from the point cloud while maintaining points representing roadside poles which is an integral step to most of the pole extraction algorithms. The high speeds at which LiDAR data is collected in rural environment also impacts the properties of the point cloud, including point density. Such conditions, although challenging, are commonly present, and so there is a need for research to explore the efficiency of detecting pole-like features in such an environment.

This paper develops an algorithm using data on a 4 km rural highway corridor where significant variation in vertical profile exists. The extraction revealed the importance of tiling LiDAR data before attempting the extraction of poles on such a large scale. Testing also revealed that, even in such a challenging environment, pole-like objects could be extracted and mapped at a decent level of accuracy. The study also points out the importance of further filtering in the post-segmentation stage when extracting poles in a rural high-speed environment.

## Literature Review

Pole-like object extraction from LiDAR in the literature generally covers two areas: tree detection for forestry applications and roadside furniture extraction for transportation applications. The methods used in previous studies range from simple scanline segmentation and shape-based clustering to more complex voxel-based approaches.

Early pole detection attempts from LiDAR used circular cross-sections (8). These methods involved taking a horizontal slice of the point cloud, creating a corresponding raster image, clustering pixels with a higher number of points, and circle fitting. Using such approaches, Bienert et al. reported 97.4% detection rates (8).

Golovinskiy et al. proposed an object detection method that involved locating, segmenting, characterizing, and classifying point clusters (9). Iterative plane fitting is first used to filter out points close to the ground. Locating candidate objects then occurs using a  $k$ -nearest neighbors (kNN) graph and a clustering algorithm. Segmentation was attempted by combining the kNN graph used to connect points into foreground objects by distance thresholds with information on the degree to which the foreground object are connected to the background. After characterizing foreground objects, objects are classified using random forests and support vector machine classifiers. Testing revealed that the algorithm was effective in locating and segmenting objects with 92% and 93% accuracy, respectively. Recognition rates were significantly lower at 58% precision and 65% recall.

Lehtomäki et al. proposed a scanline-based algorithm to extract pole-like objects from mobile LiDAR data (10). In scanline LiDAR data, poles will exist as sweeps (i.e., point groups) in each scanline. Point groups which are on top of each other in adjacent scanlines are then clustered. Clusters that constitute the same pole are merged using principal component analysis (PCA). A cluster is defined as part of a pole-like object if it meets specific geometric properties. The algorithm was tested on a 450 m straight and flat section, and the authors reported a 77.7% detection rate and 81.0% correctness rate. False-positives included pillars in buildings and different wall structures. Lamp posts were found to be the easiest to detect, with a detection rate of 93% as compared to traffic signs and tree trunks at 73.3% and 76.1%, respectively.

Pu et al. presented a method to classify MLS point clouds into three categories: ground surface, objects on the ground, and objects off the ground (11). Additionally, objects on the ground are classified into detailed groups such as traffic signs, trees, and utility poles. A surface growing algorithm taken from Vosselman et al. (12) is employed to determine the points representing the ground surface. To detect poles, the common feature of a vertical principal axis are used. Objects are divided into quartiles based on their height, and the third quartile (measured from the lowest elevation point of a given cluster) is used for further analysis. This helps omit objects such as bushes and trees when classifying ground objects into poles and non-poles. The authors reported an 87% success rate for detection of pole-like objects using their procedure.

The procedure proposed by El-Halawany and Lichti to extract poles starts by organizing the point cloud using a  $KD$  tree data structure (13). A 2D density-based segmentation is performed using a density-based clustering algorithm (DBSCAN) which finds clusters of high

density in local neighborhoods. The proximity threshold in the DBSCAN search is defined based on a utility pole radius of 25 cm. The output of the clustering is then used in a vertical region-growing procedure to extract upright objects starting from the lowest elevation object detected in the previous step as the seed for the vertical regions. To merge different vertical segments that are close enough to be considered part of the same object, segment merging is based on the horizontal distance between centroids of the vertical regions grown in the previous step. Objects are then classified using several criteria including object height range, the surface normal direction, and the largest normalized eigenvalue. The algorithm was tested using data collected on three urban streets ranging in length from 103 to 768 m, with a reported processing time of 4–6 hours. The average detection rate was 86% for the three segments and the accuracy was 97%.

Yan et al. proposed a four-step procedure to extract poles and towers from LiDAR (14). The method consists of ground filtering, unsupervised clustering, classification, and data cleaning. Filtering the ground surface from the LiDAR point cloud is done based on the statistical distribution of the points (assuming normality of ground points). This allows for a statistical skewness balancing algorithm applied to the height attribute to differentiate ground and nonground points. The paper then uses DBSCAN to cluster the height-normalized nonground points. Each cluster is classified into one of five types of poles based on a set of defined decision rules. The final stage involves using least square circle fitting algorithms on the lower 10–20% portion of the pole structure to eliminate ground points from the extracted pole object. The proposed algorithm was tested on an urban site in Toronto, Ontario, resulting in a 91% detection rate for five types of light poles and towers.

Wu et al. proposed a voxel-based method for identification of street trees from LiDAR (15). The method consists of voxelization, calculating values of voxels, searching and marking neighborhoods, extracting potential trees, and using morphological parameters to eliminating pole-like objects other than trees. It is worth noting that the voxel layer that fell 1.2–1.4 m above the ground was used to begin the neighborhood marking and searching to extract trees. The proposed algorithm was tested on two 300 m long flat urban street segments with less than 1 m difference in elevation—therefore, height normalization with respect to a ground surface was not required. This resulted in a completeness and correctness of over 98% in detection.

Cabo et al. also proposed an automatic voxel-based extraction of pole-like objects from MLS LiDAR (16). The data are first voxelized to reduce data size for processing. Each horizontal layer of the voxelization is analyzed and segmented separately and segments are then

merged to form the selected 3D features. The 2D analysis is carried out to identify pole-like candidates in three stages: segmentation of connected horizontal elements; selection of elements greater than the maximum area criteria; and selection of elements by isolation criteria. Both the second and third stages are based on the assumptions that poles have a relatively small cross-sectional area and are isolated. The results provide a set of segments associated with a Z coordinate of a candidate part of a pole. The third step involves connecting all voxel elements that share a face, edge, or vertex among all elevation layers. A minimum vertical height is set for connected groups to differentiate pole-like objects. The algorithm was successfully tested on four sites, with an average completeness of 92.3% and a correctness of 83.8%.

Lehtomäki et al. proposed an automated voxel-based method to process large MLS LiDAR point clouds and create road environment infrastructure maps (17). The proposed method consists of the following workflow: isolating nonground points from the point cloud, object segmentation, segment classification, and object location estimation. The authors use connected component labeling to perform object segmentation. Feature descriptors calculated from voxels making up a segmented object include local descriptor histograms, spin images, and general shape/point distribution attributes to apply machine learning techniques for object classification. The paper was successful in extracting and classifying trees, lamp posts, traffic signs, cars, pedestrians, and advertising boards. The authors tested their algorithm on a 900 m stretch of road in a suburban area in Espoo, Finland. In general, the authors report between 66.7% and 94.3% recall for the six defined object classes.

As evident in the review, there has been some progress in efforts to extract pole-like objects from LiDAR. The advanced techniques recently adopted have helped increase the accuracy of the results; however, in most existing studies testing was conducted on relatively short and flat urban streets. Accordingly, the feasibility of extraction on rural segments where the terrain and the length of the highway segments pose a challenge remains unknown. To address the aforementioned gaps, this paper builds on research in existing studies to develop a more robust method that can extract poles in a more challenging rural environment with rolling terrain. Specifically, the paper builds on the voxel-based approach used to perform nonground extraction since it was found to be the most effective method when dealing with significant variations in vertical alignment common in rural environments. To increase the efficiency of the voxel-based nonground extraction and reduce processing time, which is typically a challenge when dealing with long segments, the paper introduces the concept of data tiling, in which the point cloud is split into tiles or

quadrants. Moreover, when segmenting the pole-like candidates, the paper performs density-based clustering on voxels as opposed to points. This helps address the issue of the low density of points on some pole structures, which is expected in rural environments. In addition, this also helps reduce processing time while maintaining the same level of accuracy.

## Proposed Method

### Pre-processing

Pre-processing was done in three different steps. First, points of very low intensity were removed from the point cloud. These low-intensity points are generally isolated particles within the atmosphere that do not represent the road environment. The second step of pre-processing involves filtering out points within the point cloud that have elevations lower than the mean elevation of the point cloud by six standard deviations. These points are typically a result of multipath errors in which the laser beam is refracted by a surface and thus the range measurement from source to destination is severely degraded.

The final step of pre-processing involves segmenting the LiDAR point cloud into square “tiles” of data. Since the proposed procedure is intended for long LiDAR segments, it is essential that the data be segmented into manageable portions. Each “tile” of data is chosen to represent a 50 m<sup>2</sup> subset of the entire LiDAR dataset. The dataset is divided into these tiles by defining a grid of 50 m cells based on the dataset’s coordinate extents. Points are then grouped into each “tile” using a 3D histogram counting algorithm (18). This algorithm takes three vectors delineating each direction into  $N$  point groups and returns the  $i, j, k$  indices that correspond to a specific data “tile” for each point.

### Stage A: Voxelization

The first stage of the processing pipeline involves voxelization of each data tile in the point cloud. Voxelization is the process of discretizing the LiDAR point cloud into three-dimensional voxels of a certain size similar to two-dimensional pixels in a normal image.

Let  $v(i, j, k)$  denote a voxel and  $v(I, J, k)$  denote all voxels in layer  $k$ . If  $V$  represents the voxel grid system consisting of  $K$  layers, then  $V$  can be defined as the union of all voxel layers:

$$V = \bigcup_{k=0}^K k \quad (1)$$

Unlike data tiling discussed above, the voxelization of each point in the point cloud is performed based on the spatial coordinates of the point as illustrated in Figure 1.

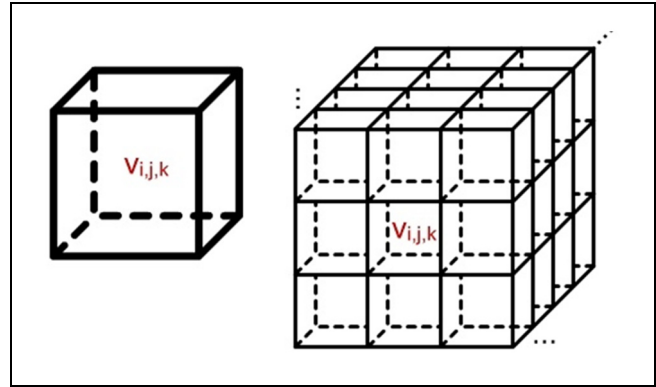


Figure 1. Voxel representation (19).

A point  $P(x, y, z)$  is assigned to a voxel  $v(i, j, k)$  as follows. If  $\Delta x$ ,  $\Delta y$ , and  $\Delta z$  denote the dimensions of a single voxel cell ( $v$ ) in the  $x$ ,  $y$ , and  $z$  directions and  $x_0$ ,  $y_0$ , and  $z_0$  denote the origin of the voxel grid ( $V$ ), then the ID of the voxel,  $v(i, j, k)$ , in which the point  $P(x, y, z)$  falls can be computed as follows:

$$i = \frac{\text{int}(x - x_0)}{\Delta x} \quad (2)$$

$$j = \frac{\text{int}(y - y_0)}{\Delta y} \quad (3)$$

$$k = \frac{\text{int}(z - z_0)}{\Delta z} \quad (4)$$

The actual voxelization of each data tile is based on the same three-dimensional histogram counting algorithm (18). The algorithm works by classifying points into bins based on the preset dimensions for individual voxel cells ( $v$ ) and information on the size of the entire voxel grid ( $V$ ).

The dimensions of the voxel are user-defined. For best overall results, it is recommended that voxel dimensions be defined based on the laser scanner properties. Since the data scanned in this study were done in scanlines that are approximately 20 cm apart, a 20 cm cell size for voxelization was used. This also ensures that points scanned from adjacent scanlines would fall in neighboring voxels, which is important when performing the connected components labeling.

### Stage B: Ground vs Nonground Filtering

After voxelization occurs, all unique foreground voxels (i.e., voxels containing points) in a certain voxel layer ( $k$ ) are found with their corresponding related points and a set of scalar and distributional geometric shape descriptors are generated. The attributes assigned to each voxel consist of a scalar geometric center and principal

components based on the distribution of points within a local neighborhood of the voxel. The local neighborhood is defined as the 26 neighbors of a voxel in three dimensions. The principal components are calculated using PCA, specifically by using singular value decomposition (SVD) to find the principal directions and their eigenvalues. Since the SVD approach requires a large enough sample, only voxels containing more than three points were considered in this step.

The PCA attributes computed for each voxel include measures of linearity, flatness, and angular measures with respect to the Cartesian directions. Voxel linearity and flatness can be surmised from the eigenvalues corresponding to the principal directions (i.e., eigenvectors) since the eigenvalues can be seen as a measure of the variance along each principal direction. Linearity can be defined by dividing the largest principal eigenvalue by the sum of the eigenvalue of the principal directions, as in Equation 5. Similarly, flatness can be defined by dividing the smallest principal eigenvalue ( $\lambda$ ) by the sum of the eigenvalues of all principal directions, as in Equation 6:

$$\text{Linearity} = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} \quad (5)$$

$$\text{Flatness} = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \quad (6)$$

The angle of the component can then be computed as follows. If  $V$  denotes an eigenvector and  $U_Z$  denotes the Cartesian  $Z$  unit vector, then the following equations are used to compute the angular measures:

$$\begin{aligned} &\text{Angle of first principal component w.r.t} \\ U_Z &= \arctan(\|V_1 \times U_Z\|, V_1 \cdot U_Z) \end{aligned} \quad (7)$$

$$\begin{aligned} &\text{Angle of third principal component w.r.t} \\ U_Z &= \arctan(\|V_3 \times U_Z\|, V_3 \cdot U_Z) \end{aligned} \quad (8)$$

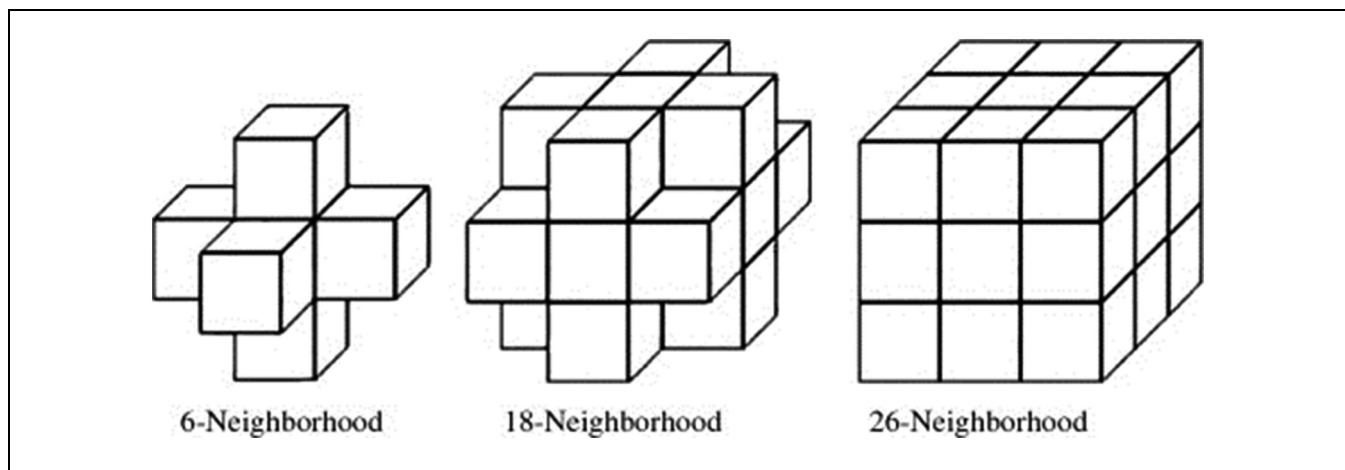
Both attributes of flatness and angle of the third eigenvector with the unit  $z$  vector help filter voxels that are expected to have high flatness and close to vertical third principal component (i.e., voxels representing the ground surface).

Once ground surface voxels are filtered out, all remaining voxels will then represent nonground objects. Data points must be voxelized into a small enough voxel size for the third principal component of potential road segments to be close to vertical, especially considering many rural highways have large vertical curves. Therefore, the choice of a 20 cm voxel size.

### Stage C: Connected Component Labeling

After voxelization and ground filtering, a binary 3D image of the nonground voxels within each data tile is generated with foreground voxels represented as value "1" and background voxels represented as value "0". Connected component labeling (CCL) is then performed. This involves placing voxels within close proximity under a single label (see Figure 2). In this paper, CCL is done based on 26-connectivity, which means that, for each central voxel, adjacent voxels in the 3D space (up to the 26 neighbors) are considered when grouping voxels.

Once connected components are found, they are reported as potential objects in each voxel layer. These potential clusters of objects are typically only part of a larger object. For instance, different parts of tall objects such as poles could fall in voxel layers that are not adjacent and thus segment merging is required. Two-dimensional DBSCAN clustering is used to perform segment merging on cluster centroids. This filter ensures that changes in the 2D distance between cluster centroids in



**Figure 2.** Central voxel neighborhood (in this paper, CCL is based on 26-neighborhood connectivity) (20).

adjacent voxel layers are not significant, while at the same time merging clusters in different voxel layers with insignificant changes in 2D distance. Employing these two criteria, segment merging is performed which enables merged segments to be representative of meaningful physical objects in the road environment.

#### Stage D: Pole Identification and Classification

This stage involves the computation of shape descriptors for each of the merged segments (i.e., objects). These descriptors are then used to further classify detected objects into poles and non-poles. In addition to the general geometric descriptors such as number of points and the height of objects, PCA is used to compute the same attributes as for the individual voxels but now on segmented objects. This includes linearity, flatness, and two angle measures. Streetlight and utility poles are thin vertical objects with a distinct range in the first principal component (i.e., high linearity) and first and third principal component angles with respect to the Cartesian  $z$  direction. Specifically, the following decision rules were used in the classification:

For east two-thirds of the test segment:

1. linearity between 0.7 and 0.8
2. flatness greater than 0.975
3. surface normal angle between  $85^\circ$  and  $95^\circ$
4. first principal component angle less than  $75^\circ$

For west one-third of the test segments:

1. linearity between 0.8 and 0.9
2. flatness greater than 0.9
3. surface normal angle between  $65^\circ$  and  $95^\circ$

It is worth noting here that different filters for different portions of the 4 km test segment were used since parts of the segment included poles with incomplete structure (this is discussed further in the Results and Discussion section).

#### Case Study

The algorithm developed in this paper was tested on LiDAR data collected on a 4 km segment along Highway 20 in Alberta, Canada. The test segment in this paper was much longer (4 km) than in previous studies, and included high variation in vertical alignment ( $\Delta z = 796$  m). A longer segment length with varying vertical alignment was purposely chosen to test the algorithm since it represents a realistic rural setting. The next two

paragraphs provide more details on LiDAR data collection procedure and the test segment.

#### LiDAR Data

LiDAR data were collected using REIGL's VMX 450 Laser Scanning System. The RIEGL VMX-450 system uses two VQ-450 scanners along with the GNSS/IMU units to collect the data. The laser scanners are symmetrically configured on the left and right sides, pointing toward the rear of the vehicle at a heading angle of approximately  $145^\circ$ . The VQ-450 scanner has a scan rate reaching 1.1 million points per second and a speed of 400 lines per second (13). The density of the points on a scanned object depends on the range, and the speed of the data collection truck; provincial surveys conducted at 90 km/h result in point densities on the pavement surface of 150–1000 points/m<sup>2</sup> (14).

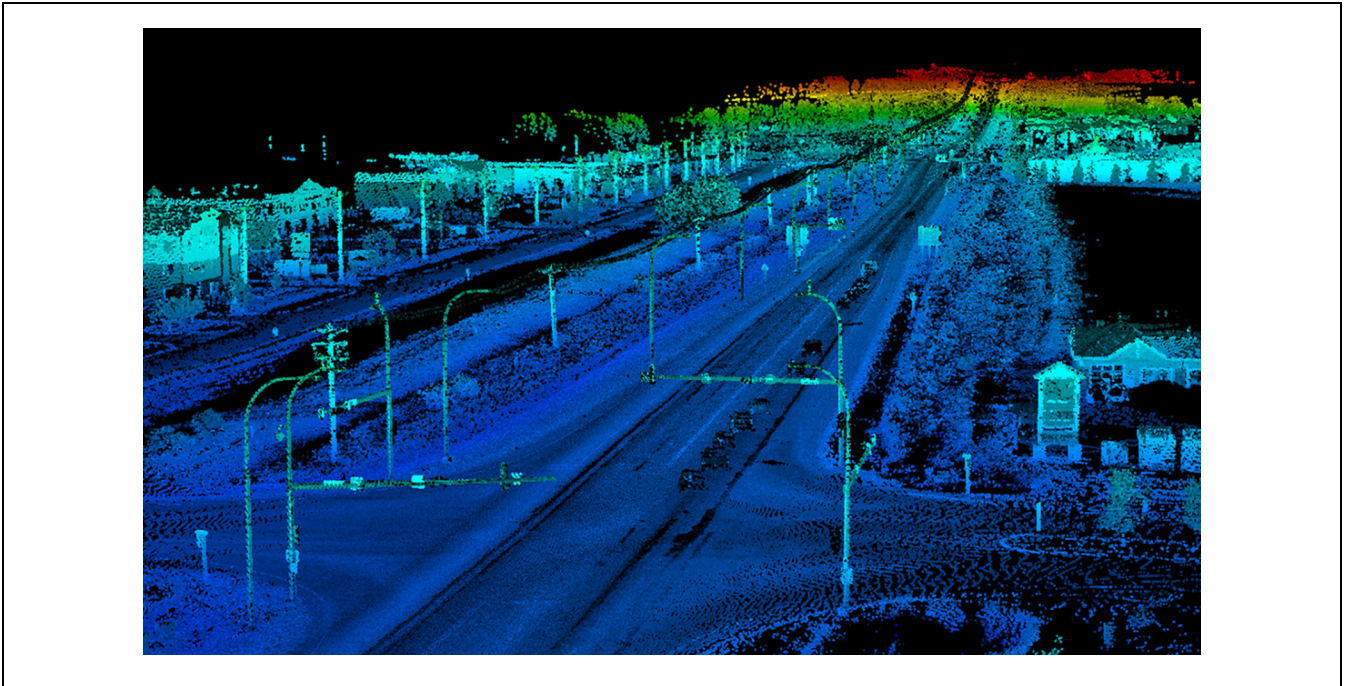
#### Test Segment

The analyzed portion of Highway 20 is an undivided highway located close to Sylvan Lake (west of Red Deer). A portion of the test segment is shown in Figure 3. The speed limit on the segment is 100 km/h; however, the speed does drop for a 1 km portion of the segment that runs close to the town of Sylvan Lake. Vegetation along the sides of the road is moderate. In addition to the drastic change in vertical alignment along the 4 km section, there is also a reverse horizontal curve on the segment and a rural intersection. Ground truth data used in the validation of the extracted information was collected through manual exploration of the test segment.

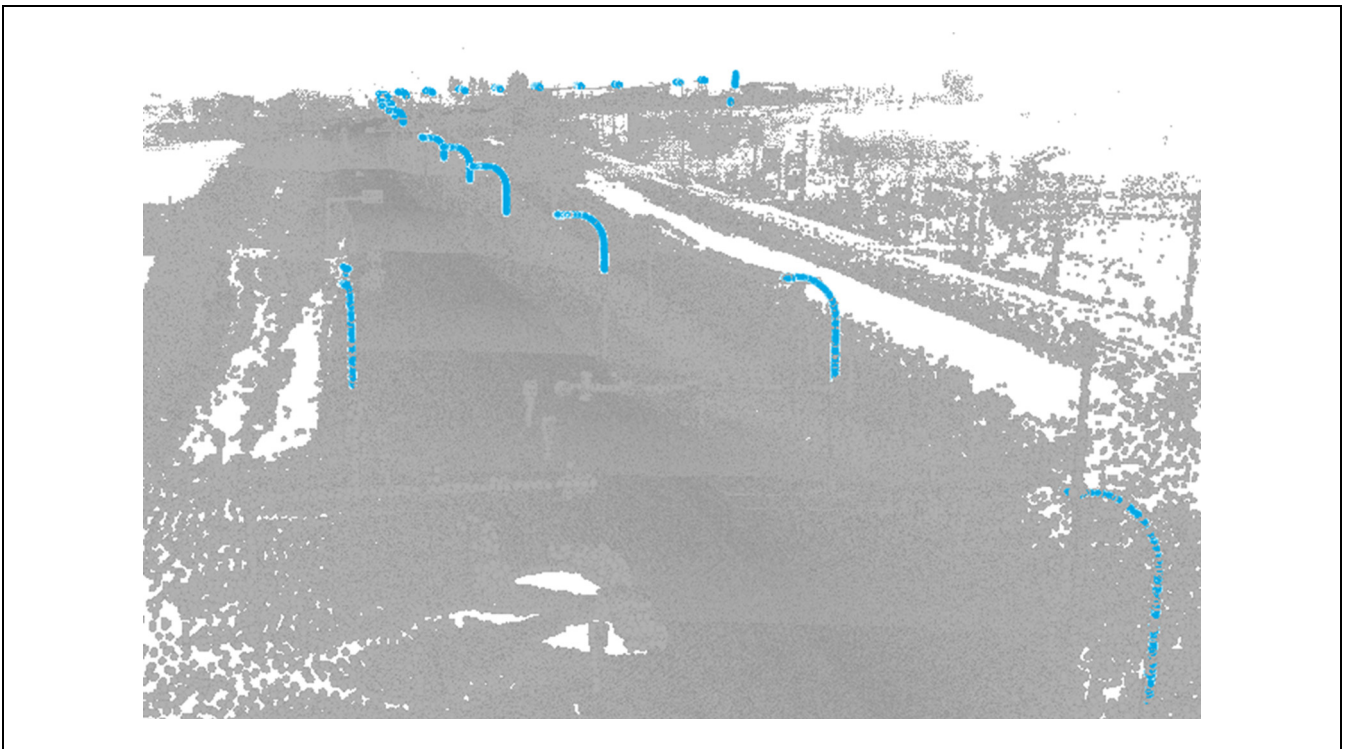
#### Results and Discussion

Before presenting the results, it is worth emphasizing that the test segment used in this paper was 4 km long with rolling terrain. The length of the segment increases the number of nonground objects from which poles are filtered. On the segment tested in this study, the number of objects detected before classification was 24,767 nonground objects. This includes signs, trees, building facades, vegetation, and poles. Out of those 24,767, only 41 poles existed; as a result, the likelihood of other objects having the same properties as poles is much higher when compared to analyzing a short 100 m segment where fewer objects exist. Moreover, the rolling terrain on the test segment causes a difference in elevation of 796 m. This makes extracting the nonground surface using voxelization of the point cloud extremely challenging. Despite that, nonground classification was highly accurate.





**Figure 3.** Highway 20.



**Figure 4.** LiDAR points representing detected poles.

Figure 4 shows points in the LiDAR point cloud representing the detected poles. Figure 5a shows a map of all the detected poles along the 4 km segment with the ground truth shown in Figure 5b. Similarly, Figures 5c and d show

the detected poles and the ground truth for a 1 km test segment that runs adjacent to Sylvan Lake. It is noticeable from the figures that the algorithm was more effective in classifying poles in the section closer to Sylvan Lake town.





**Figure 5.** Pole distribution along the test segment: (a) ground truth (4 km segment); (b) true-positives (4 km segment); (c) ground truth (1 km segment—low speed); (d) true-positives (1 km segment—low speed).

The extracted information can be mapped on the highway segment with pinpoint accuracy, which facilitates measuring the density of light poles on a road segment along with their offsets from the road. These two critical performance measures were found to be the most significant factors in studies that analyzed the effects of lighting standards on safety.

To numerically assess the validity of the results, three metrics were calculated. The metrics were precision, detection rate, and accuracy. These were calculated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (9)$$

$$\text{Detection rate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (10)$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}} \quad (11)$$

where TP and TN denote the number of true-positives (actual poles detected) and the true-negatives (actual non-poles detected). FP denotes the number of false-positives (i.e., the number of objects detected as poles which were not actually poles) and FN denotes the

number of false-negatives (i.e., the number of objects that were actually poles but that were not detected by the algorithm as poles).

Accuracy measures how effective the algorithm is in the valid classification of both true-positives and true-negatives. This measure is also known as quality and provides a compound performance metric that balances detection rate and precision (21). Detection rate, also known as completeness, measures how effective the algorithm is in the valid classification of true-positives only. Finally, precision, also known as correctness, measures how successful the algorithm is in applying the classification filters.

As evident in the table, both precision and accuracy are relatively high for both the entire segment and the section closer to Sylvan Lake, which points to the robustness of the proposed algorithm in extracting over-ground objects. The detection rate, however, is high for the portion close to Sylvan Lake but relatively low for the entire segment. This indicates that the classification procedure is not as efficient as the extraction (i.e., poles are being detected but some poles are getting misclassified as other objects). This is mainly caused by two things, both of which have been reported in previous research (22). First, there are some cases in which the poles are too close to

**Table 1.** Result Validity Assessment

Metric	4 km segment	Sylvan Lake (1 km segment)
Precision (%)	68	78.6
Detection rate (%)	49	78.6
Accuracy (%)	98	95.4

other objects such as vegetation (bushes) or signs. This results in mixed pixels between the two objects (the pole and other objects) and, as a result, the objects get segmented into one object in the CCL process.

The other reason some poles were misclassified is because portions of the pole structure were missing (i.e., the pole structure was incomplete). The incomplete structure of poles causes the CCL to segment the pole into two objects. The CCL works by segmenting adjacent foreground voxels into a single object; however, when one of the adjacent voxels does not contain points, this leads to the voxel groups being classified as two different objects. This causes an issue when classifying pole structures since a pole that is divided between two segments or only has the top portion detected does not have the same geometric properties as a complete pole that belongs to a single cluster.

Although DBSCAN clustering was used in an attempt to regroup those unconnected segments, this was not always possible. This issue was more prevalent on the portion of the highway further away from the town of Sylvan Lake, as evident in the detection rates shown in Table 1. Therefore, it could be a matter of the data collection truck traveling at higher speeds away from the city bounds causing a reduction in point density. Object occlusion could also be a reason why some pole structures were incomplete. To minimize the effects of this, different filtering thresholds were used to classify poles in different regions of the test segment. The incomplete structure of a pole might also affect the ability to measure dimensions of the detected pole, although this is beyond the scope of this paper.

## Conclusions and Future Research

This paper proposes an algorithm to automatically extract and classify pole-like objects using LiDAR point cloud data on rural highways with rolling terrain. The algorithm involves breaking the data down into manageable data tiles. Each data tile is then voxelized and voxel attributes are used to classify the point cloud into ground and nonground points. After voxelization, CCL is used to segment different parts of the nonground surface into potential objects. Further clustering of objects is then done and the objects are classified into poles and non-

poles based on their geometric attributes. The proposed algorithm is tested on a 4 km rural highway segment in Alberta, Canada. In general, the results were accurate; however, misclassification of poles did exist owing to low point density and pole occlusion by other objects. This paper illustrates the importance of accounting for the road type and its terrain when extracting information from LiDAR. The results also indicate that more robust segmentation may be required when extracting poles in a rural high-speed environment to account for the lower point density caused by the high data collection speeds. To address this issue, future research might consider employing machine learning techniques in the segmentation stage. Similarly, ellipsoidal region-growing algorithms could also help overcome the limitations associated with the CCL when dealing with discontinuity in pole structure. The proposed procedure could also be replicated on a segment where LiDAR is collected using a phase-based scanner instead of the time-of-flight scanner used in this paper. Phase-based scanners often result in a more dense point cloud, albeit the range of the data collection is shorter (23). This could help increase the density of points on the pole-like objects.

## Acknowledgments

The authors would like to thank Alberta Transportation for sponsoring and providing data for this study. Acknowledgments are also extended to Alberta Innovates and Alberta Advanced Education for financial support.

## Author Contributions

The authors confirm contribution to the paper as follows: All authors contributed to the study conception and design; algorithm development—Suliman A. Gargoum and James C. Koch; analysis, testing, and interpretation of results—Suliman A. Gargoum and James C. Koch; all authors contributed to the drafting manuscript preparation. All authors reviewed the results and approved the final version of the manuscript.

## References

1. Zegeer, C. V., and M. R. Parker, Jr. Effect of Traffic and Roadway Features on Utility Pole Accidents. *Transportation Research Record: Journal of the Transportation Research Board*, 1984. 970: 65–76.
2. El Esawey, M., and T. Sayed. Evaluating Safety Risk of Locating Above Ground Utility Structures in the Highway Right-of-Way. *Accident Analysis & Prevention*, Vol. 49, 2012, pp. 419–428.
3. Gargoum, S. A., K. El-Basyouny, J. Sabbagh, and K. Froese. Automated Highway Sign Extraction Using Lidar Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2017. 2643: 1–8.
4. Guan, H., J. Li, Y. Yu, C. Wang, M. Chapman, and B. Yang. Using Mobile Laser Scanning Data for Automated

- Extraction of Road Markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 87, 2014, pp. 93–107.
5. Holgado-Barco, A., B. Riveiro, D. González-Aguilera, and P. Arias. Automatic Inventory of Road Cross-Sections from Mobile Laser Scanning System. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 32, No. 1, 2017, pp. 3–17.
  6. Kumar, P., C. P. McElhinney, P. Lewis, and T. McCarthy. An Automated Algorithm for Extracting Road Edges from Terrestrial Mobile LiDAR Data. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 85, 2013, pp. 44–55.
  7. Castro, M., L. Iglesias, J. A. Sánchez, and L. Ambrosio. Sight Distance Analysis of Highways Using GIS Tools. *Transportation Research Part C: Emerging Technologies*, Vol. 19, No. 6, 2011, pp. 997–1005.
  8. Bienert, A., S. Scheller, E. Keane, F. Mohan, and C. Nugent. Tree Detection and Diameter Estimations by Analysis of Forest Terrestrial Laserscanner Point Clouds. *Proc., ISPRS Workshop on Laser Scanning 2007 and Silvi-Laser 2007*, Espoo, 2007, pp. 50–55.
  9. Golovinskiy, A., V. G. Kim, and T. Funkhouser. Shape-Based Recognition of 3D Point Clouds in Urban Environments. *Proc., 2009 IEEE 12th International Conference on Computer Vision*, Kyoto, Japan, IEEE, New York, 2009, pp. 2154–2161.
  10. Lehtomäki, M., A. Jaakkola, J. Hyypä, A. Kukko, and H. Kaartinen. Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data. *Remote Sensing*, Vol. 2, No. 3, 2010, pp. 641–664.
  11. Pu, S., M. Rutzinger, G. Vosselman, and S. O. Elberink. Recognizing Basic Structures from Mobile Laser Scanning Data for Road Inventory Studies. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 66, No. 6, 2011, pp. S28–S39.
  12. Vosselman, G., B. G. Gorte, G. Sithole, and T. Rabbani. Recognising Structure in Laser Scanner Point Clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 46, No. 8, 2004, pp. 33–38.
  13. El-Halawany, S. I., and D. D. Lichti. Detection of Road Poles from Mobile Terrestrial Laser Scanner Point Cloud. *Proc., 2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM)*, Xiamen, China, IEEE, New York, 2011, pp. 1–6.
  14. Yan, L., H. Liu, J. Tan, Z. Li, H. Xie, and C. Chen. Scan Line Based Road Marking Extraction from Mobile LiDAR Point Clouds. *Sensors*, Vol. 16, No. 6, 2016, p. 903.
  15. Wu, B., B. Yu, W. Yue, S. Shu, W. Tan, C. Hu, Y. Huang, J. Wu, and H. Liu. A Voxel-Based Method for Automated Identification and Morphological Parameters Estimation of Individual Street Trees from Mobile Laser Scanning Data. *Remote Sensing*, Vol. 5, No. 2, 2013, pp. 584–611.
  16. Cabo, C., C. Ordoñez, S. García-Cortés, and J. Martínez. An Algorithm for Automatic Detection of Pole-Like Street Furniture Objects from Mobile Laser Scanner Point Clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 87, 2014, pp. 47–56.
  17. Lehtomäki, M., A. Jaakkola, J. Hyypä, J. Lampinen, H. Kaartinen, A. Kukko, E. Puttonen, and H. Hyypä. Object Classification and Recognition from Mobile Laser Scanning Point Clouds in a Road Environment. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 54, No. 2, 2016, pp. 1226–1239.
  18. Luong, B. N-Dimensional Histogram. MATLAB Central File Exchange, 2009. Updated August 2011. <https://www.mathworks.com/matlabcentral/fileexchange/23897-n-dimensional-histogram>. Accessed May 2017.
  19. Zirbes, R. Scientific Visualization: Volume Surface Rendering. <http://johnrichie.com/V2/richie/isosurface/volume.html>. Accessed October 2017.
  20. Kinkingnehun, S., E. Volle, M. Péligrini-Issac, J. L. Gormard, S. Lehéricy, F. du Boisguéheneuc, S. Zhang-Nunes, D. Sosson, H. Duffau, and Y. Samson. A Novel Approach to Clinical–Radiological Correlations: Anatomic–Clinical Overlapping Maps (AnaCOM)—Method and Validation. *Neuroimage*, Vol. 37, No. 4, 2007, pp. 1237–1249.
  21. Rutzinger, M., F. Rottensteiner, and N. Pfeifer. A Comparison of Evaluation Techniques for Building Extraction from Airborne Laser Scanning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 2, No. 1, 2009, pp. 11–20.
  22. El-Halawany, S. I., and D. D. Lichti. Detecting Road Poles from Mobile Terrestrial Laser Scanning Data. *GIScience & Remote Sensing*, Vol. 50, No. 6, 2013, pp. 704–722.
  23. Guan, H., J. Li, S. Cao, and Y. Yu. Use of Mobile LiDAR in Road Information Inventory: A Review. *International Journal of Image and Data Fusion*, Vol. 7, No. 3, 2016, pp. 219–242.

*The Standing Committee on Geographic Information Science and Applications (ABJ60) peer-reviewed this paper (18-04591).*