

Article

## Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data

Matti Lehtomäki \*, Anttoni Jaakkola , Juha Hyypä , Antero Kukko and Harri Kaartinen

Finnish Geodetic Institute, Department of Remote Sensing and Photogrammetry, P.O. Box 15, 02431 Masala, Finland; E-Mails: anttoni.jaakkola@fgi.fi (A.J.); juha.hyypa@fgi.fi (J.H.); antero.kukko@fgi.fi (A.K.); harri.kaartinen@fgi.fi (H.K.)

\* Author to whom correspondence should be addressed; E-Mail: matti.lehtomaki@fgi.fi;  
Tel.: +358-9-295-55-226; Fax: +358-9-295-55-200.

Received: 9 December 2009; in revised form: 9 January 2010 / Accepted: 8 February 2010 /  
Published: 26 February 2010

---

**Abstract:** Accurate road environment information is needed in applications such as road maintenance and virtual 3D city modelling. Vehicle-based laser scanning (VLS) can produce dense point clouds from large areas efficiently from which the road and its environment can be modelled in detail. Pole-like objects such as traffic signs, lamp posts and tree trunks are an important part of road environments. An automatic method was developed for the extraction of pole-like objects from VLS data. The method was able to find 77.7% of the poles which were found by a manual investigation of the data. Correctness of the detection was 81.0%.

**Keywords:** mobile mapping; vehicle-based laser scanning; feature extraction; utility pole; traffic sign; tree trunk

---

### 1. Introduction

Detailed road environment information is needed for an increasing number of different applications, such as noise modelling, road safety, road maintenance, location-based services, driver assistance systems and car and pedestrian navigation. Pole-like objects are common in road environments. These include different kinds of utility poles, such as traffic lights, signals and signs, lamp posts, pylons and flagpoles. Additionally, many trees have a pole-like trunk. Inventory of different kinds of utility poles in a street or a highway can be used, for example, in street planning and maintenance. Trees are also

important because of the visual obstruction they incur. In personal navigation applications different poles are important because they are often visible to the observer and stand out from the environment. That is why it is important to include them in the virtual 3D city models. Poles can also be used as reference targets in different applications. The accuracy of (semi-)automatic georeferencing of data produced by a mobile mapping system could be improved with pole-like reference targets. Also, the positioning of a car in future driver assistance systems could be based on mapped poles in road environments [1]. Previous studies concentrating on extracting pole-like objects from road environments include [1-6]. In [1-3] the detection was based on data collected by a mobile laser scanner, while in [4] laser and image data was fused to detect road signs. In [5] a method for the detection of vertical traffic infrastructure using video sequences was introduced. In [6] neural networks were used to detect utility poles from image sequences.

### 1.1. Overview of Mapping Technologies

The modelling technology for collecting 3D road environment information can be divided into use of airborne, terrestrial and land-based mobile mapping (hereinafter mobile mapping) techniques. In this paper laser-based mapping systems are reviewed since they allow a more automatic data processing than image-based techniques. Airborne laser scanning (ALS) has been used increasingly for city modelling, digital terrain model generation and forest inventory [7-11]. In ALS large areas can be covered cost-effectively at a time. The distance of the airborne scanner to the targets in the ground is usually between several hundred metres and few kilometres. The acquired point density (1–20 pts/m<sup>2</sup>) is high enough, for example, for rough extraction of building roof structures and outlines. However, it is not high enough for more detailed modelling: building façades are difficult to model due to viewing direction and details in the buildings, such as balconies and windows, are missed. Point density in ALS is not good enough for a more detailed road surface modelling. For example, wearing of the road surface and holes cannot be detected and modelled. Due to viewing angle and point density, detection of pole-like objects is difficult. Ahokas *et al.* [12] found that with 400 m flight altitude (resulting in about 10 pts/m<sup>2</sup>) I-, Γ- and T-shaped street lamps can be detected quite well (detection rates 63%, 91% and 100%) with a Toposys scanner having 1-mrad-wide beam. However, from 800 metre altitude detection rates of I- and Γ-shaped lamps were clearly lower (7% and 52%).

Static terrestrial laser scanning (TLS) can produce dense point clouds (up to few hundred thousands of pts/m<sup>2</sup> from a range of 10 m) from which environments can be modelled accurately. For example, buildings [13-15] and trees [16,17] can be modelled in high detail. However, a relatively small area can be mapped with high accuracy. Since several scans are needed to cover larger areas and to prevent occlusion, the TLS based modelling is work intensive. Transportation and positioning of the equipment, design of different scanner positions to minimise occlusions and matching overlapping scans need a lot of manual work.

Vehicle-based laser scanning (VLS) can be considered to fill the gap between ALS and TLS. Here VLS is defined as a laser scanning system that is mounted on a vehicle that can be, for example, a car or a boat and in which measurements are made while the platform is moving. So-called stop-and-go systems are not included because they are inherently similar to TLS. VLS can produce relatively dense point clouds (up to several hundreds of pts/m<sup>2</sup> at 10 m range) fast and efficiently from large areas. VLS

is a special case of a more general land-based mobile mapping system (MMS), a multi-sensor system that incorporates different kinds of positioning, navigation and data collection sensors. They are all mounted on a rigid platform that can be, e.g., a car or a van. Typically, the positioning and navigation system consists of a GPS (Global Positioning System) receiver and an inertial measurement unit (IMU). The data acquisition sensors can be, e.g., laser scanners, digital cameras or video cameras and the type of sensors used depends on the type of data needed [18]. Use of laser scanners in MMS offers a possibility to collect direct geometric 3D information from the road environment. Dense point clouds can be achieved which makes it possible to construct a detailed 3D model of the road environment. Road and ground surface, building façades, trees, road geometry, kerbstones, road markings and cracks in the road surface can be retrieved and modelled [19-23].

Mobile mapping technology has developed since the end of 1980s [24]. Most applied imaging sensors in MMS have been video and digital cameras. In the beginning of the 21st century laser scanners also started to be incorporated into the systems. They have been mainly used with cameras but also systems based purely on laser have been developed. Reviews of mobile mapping technology and different systems can be found in [18,25,26]. A review of laser-based mobile mapping systems can be found in [24]. Currently there are two prominent commercial laser-based mobile mapping systems in the market. Optech's Lynx Mobile Mapper V200's platform is adaptable to any vehicle with standard roof racks. It consists of 1–2 scanners with a maximum 200 kHz point repetition frequency (PRF) and maximum range of 200 metres. 3D laser mapping company's latest model, StreetMapper 360, uses Riegl's VQ-250 scanners with a PRF of 300 kHz and maximum range of 300 m.

### *1.2. Overview of Methods Related to Extraction of Pole-Like Objects*

Various methods have been developed for the processing of 3D point clouds collected, e.g., with ALS or TLS. Some of them are readily applicable for VLS data but because of differences in point density and spacing, scanning geometry and point measurement accuracy, new methods have to be developed specifically for VLS. View angle from the measuring device to the targets in ALS is very different in VLS compared to ALS. In VLS, measurements are made from a moving platform and limited accuracy of GPS and IMU produce errors which cause that the 3D point measurements are less accurate than with TLS in which the measurements are made on a (static) tripod. At the moment, there does not exist a reported study of the statistical accuracy of any of the pole detection methods developed for VLS data.

Most of the poles and tree trunks have a circular cross-section and thus a cylindrical or conical shape. Therefore, they could be detected by automatic cylinder detection methods which have been developed in large numbers for range data (see, e.g., [27-35]). Methods utilise, e.g., Hough transform [30], Ransac [31-33] and least squares fitting [28,29]. Automatic detection of tree trunks from laser data has also been studied intensively in recent years (see e.g., [36-43]). Many of the methods utilise the fact that the horizontal cross-section of the trunk is close to circular and they try to detect circle arcs from horizontal slices of point clouds [36-39]. It is also possible to analyse the local point distribution around each point. This way data points can be classified into different classes such as scattered (grass and tree canopy), linear (thin objects like small tree trunks) and surface (ground) [44].

This approach has also been used in the tree trunk detection [43]. It is also possible to utilise contextual information by using Conditional Random Field models [45].

In VLS, pole detection has been studied in [1-3]. In [3] mobile mapping data consisted of vertical scan lines and vertical poles were detected by extracting vertical line segments from individual scan lines. Detection was limited to poles that were in front of buildings and on the ground. In [2] a method for the extraction of traffic signs and signals was introduced. Points were clustered to different objects and clusters were projected to a vertical plane. Finally, poles were detected based on their elongated form in the projection using the eigenvalue decomposition of the covariance matrix. In the method, a pole does not have to be vertically aligned. However, if a pole is part of some bigger object which is not pole-like (a tree, a traffic sign with a large board) it is not extracted but considered as being part of the object. The method is fully automatic only if there are kerbstones on both sides of the road which are used for the estimation of the road surface. In [1] an extraction method for vertical poles was introduced. In the method, a pole is expected to have an almost empty region around it. The structure is analysed in cylindrical stacks and if enough stacked cylinders are found a pole is extracted.

In this study, a method for the automatic extraction of vertical pole-like structures from VLS data is developed. The method applies the ideas presented in [31,35] in which individual scan lines, true or artificial, are segmented and possible sweeps of cylinders are extracted from the segments. In [31] geometric models (circles or ellipses) are fitted to the point segments to classify sweeps. After that the circles or ellipses are clustered to form cylinders. In [35] overlapping point segments are searched for from adjacent horizontal scan lines to find vertical elongated objects. Circle models are fitted to the point segments to find cylinders and to estimate their axes and radii. Our method is similar to [35] except that no geometric models are fitted in any phase of the extraction process. The length of the point segment is used to separate possible sweeps from other segments and clustering is done by searching segments that are approximately on top of each other in adjacent scan lines. The found clusters are merged if needed by using a new method that utilises the linear structure of the poles. Because no geometric model is applied, the found clusters are not final features but intermediate candidates for poles. The method detects vertical linear structures from the clusters using similar ideas as in [43,44]. However, instead of classifying individual points in unstructured point clouds, our method utilises the scan line structure of the laser data. The method is somewhat similar to [1] in which the structure is analysed in cylindrical stacks.

The originality of the new method compared to previous pole detection methods in VLS is its generality. Firstly, no prior assumptions about the positions of the poles are made—the only things limiting the positions are the maximum measurement range of the laser scanner and point density which decreases as the distance to the road increases. Secondly, poles attached to other objects, like in traffic signs (sometimes many poles in one sign) and trees are all detected as separate objects. This is important, e.g., in detailed modelling of the objects. The number of extracted poles (features) will also be higher which is important for positioning of a vehicle in future driver assistance systems [1].

### *1.3. Objectives of the Study*

The objective of this study is to develop automatic methods for the detection of vertical pole-like structures in road environments using vehicle-based laser scanning. A major contribution of this study

is to test experimentally and analyse statistically a pole detection method for VLS for the first time. The quality of the VLS data from the point of view of pole detection is also analysed shortly. Reference data collected from the test field is used to evaluate the performance of the developed method. The high level of automation is important because of the huge amounts of data produced by VLS.

**Table 1.** Main properties of the SPAN's IMU (HG1700 AG11).

Property	Value
Gyro input range	$\pm 1,000$ deg/s
Gyro rate bias	1.0 deg/h
Gyro rate scale factor	150 ppm
Angular random walk	0.125 deg/hr
Accelerometer range	$\pm 50$ g
Accelerometer linearity	500 ppm
Accelerometer scale factor	300 ppm
Accelerometer Bias	1.0 mg

## 2. Material

### 2.1. FGI Roamer

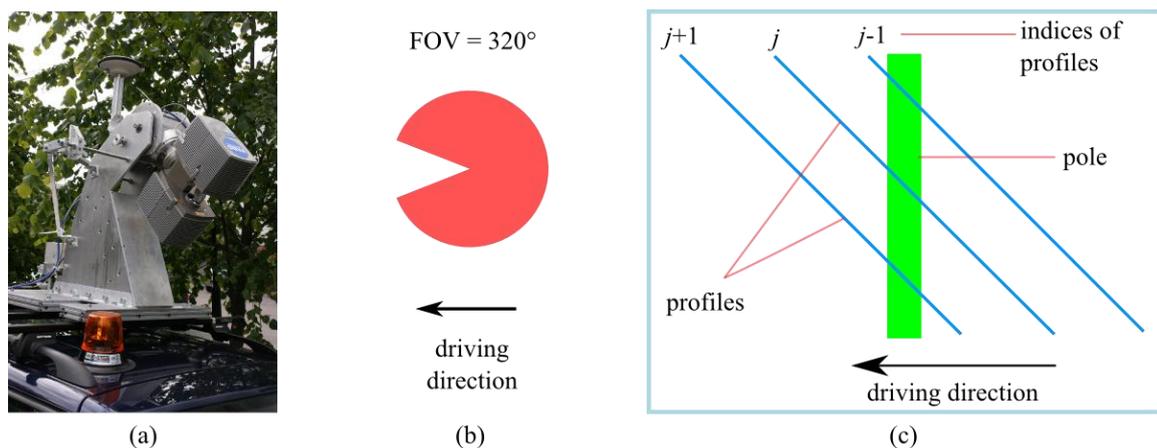
The Finnish Geodetic Institute (FGI) has been developing a mobile mapping system called FGI Roamer (Road Environment Mapper) since 2003. The main goal in the research has been to develop a system that collects data for the development of fast automatic extraction algorithms of the road environment. Roamer consists of a rigid platform on which a navigation system and a laser scanner are mounted (Figure 1a). The platform can be mounted on a vehicle using roof racks. The navigation system is NovAtel's SPAN (Synchronized Position Attitude & Navigation, Table 1). It integrates GPS and inertial data and can be used in applications which require better functionality and reliability than traditional stand-alone GPS. In this study, the Faro LS 880HE80 laser scanner was used for the range measurements (Table 2). The scanner contains a rotating mirror that deflects the laser beam and gives a 320 degree field of view (FOV) in the scanning plane. Faro LS is based on phase difference technique. The PRF of the scanner is 120 kHz and maximum measurement distance approximately 50 m in practise. Recently, the scanner has been updated to the FARO Photon<sup>TM</sup> 120 which has maximum PRF of 976 kHz and maximum measurement distance of 150 m.

**Table 2.** Basic properties of the FARO LS 880HE80 laser scanner.

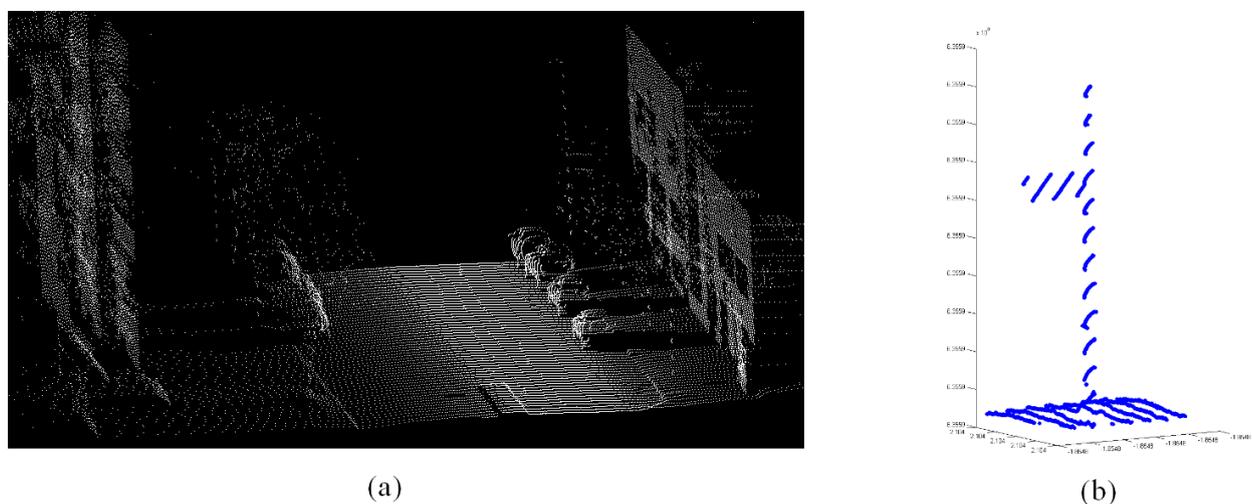
Property	Value
Maximum range	~76 m
Measurement rate	120 kHz
Angular range in scanning plane	320°
Beam divergence	0.2 mrad, 3 mm at aperture
Angular resolution (15 Hz mirror frequency)	0.045°
Ranging error	$\pm 3$ mm, linearity e.

Faro LS can be set to so-called tunnel mode in which it does 2D scans or profile measurements. The third dimension comes from the motion of the vehicle. Data consists of profiles or scan lines that are formed when the scanner’s mirror rotates one round. In its normal position, the scanning plane is vertical and perpendicular to the driving direction. The scanner can also have different tilted positions in which the scanning plane is tilted backwards with respect to vehicle motion. By using a big enough tilting angle it is possible to measure the road surface. Due to the wide FOV in the scanning plane, road surroundings can be measured at the same time (Figures 1b and 2a).

**Figure 1.** (a) FGI Roamer. (b) FOV of the system from upward direction. (c) Scanning geometry and a pole from side perspective (all figures taken or modified from [46]). In all figures the scanning plane is tilted backwards.



**Figure 2.** (a) Example of MMS data collected with FGI Roamer. (b) Part of a pole from the same data set with a traffic sign board attached to it. The road and its surroundings are mapped simultaneously and several sweeps are achieved from the pole when the scanning plane is tilted backwards.



With a tilted position narrow vertical objects like poles and tree trunks can have multiple sweeps from adjacent profiles (Figures 1c and 2b). The distance between profiles depends on driving speed and mirror frequency. Profile repetition frequency, *i.e.*, the mirror frequency can be set between 3 and 61 Hz. Due to the high point repetition frequency the system has high angular resolution in the

scanning plane. In the data set used in this study mirror frequency was 15 Hz which gives a point spacing of about 2.5 cm at 30 metres distance. For a more detailed description of the Roamer system, the reader is referred to [46,47].

## 2.2. Test Site and Collection, Pre-Processing and Quality of Data

The developed pole detection method was tested in a suburban area. Data was collected by the Roamer system and the test site consisted of about 450 metres long nearly straight stretch of road with no hills. The test site was located in the Espoonlahti district of Espoo. There were numerous traffic signs, lamp posts and trees with a pole-like trunk in the test site making it suitable for testing of the automatic pole detection method. Driving speed varied between 0 and 33 km/h and it was approximately 20 km/h on average. The tilting angle of the scanner was 135 degrees. Data was collected by driving across the same area in two opposite directions but in this study data from only one driving direction was used. The data set consisted of 1,000 profiles each containing about 7,000 points and thus the size of the whole data set was approximately 7 million points.

Based on navigation data and different transformations, the measured points were georeferenced into the WGS84 coordinate system. After that the data was transformed again using the principal component analysis (PCA) which is also known as the Karhunen-Loève decomposition (see e.g., [48]). PCA was applied to the whole data set. The direction of largest variance, *i.e.*, first principal component (PC) was approximately parallel to the road and driving direction and was chosen as the *y*-axis. The second PC represented approximately the other horizontal direction which was perpendicular to the road direction. It was chosen as the *x*-axis and thus the *xy*-plane represented the horizontal road and ground surface approximately. The *z*-axis was the last eigenvector. It was approximately perpendicular to the road surface and thus represented the vertical direction. All vertical poles would be nearly parallel to the *z*-axis after the applied coordinate transformation. PCA was possible to apply in this case because the test site did not include corners and the terrain was flat. In more general case including hills and corners, a map projection can be applied in which the poles will also be parallel to *z*-direction.

In addition to distance measurements and corresponding angle values, Faro LS also provides information on the relative positions of the measured points. Each profile has its own unique index and each measurement can be connected to a certain profile. When the car moves straight ahead, profiles with adjacent indices are adjacent also in the 3D space (Figure 1c). The points' relative positions on the profile are also known. With 135 degrees tilting angle profile spacing is the same in driving and vertical directions. With the maximum driving speed of 33 km/h it was approximately 61 cm. Average spacing over all profiles was 37 cm.

Absolute accuracy of the MMS data is determined mainly by the accuracy of the navigation system. The main causes for errors are the GPS outages; for example, buildings, bridges, tunnels, and high vegetation may prevent a good connection to satellites. Although inertial data compensates this error, its accuracy decreases quickly as a function of time. Thus, longer outages can cause bigger errors. While the accuracy of navigation data is poor, the relative accuracy between adjacent data points remains static and is in the order of centimetres. Because poles are not accurately vertical in reality, the sweeps of poles are not accurately on top of each other. Most of the poles have a circular cross-section

and this fact could be utilised in the automatic detection methods. However, only about 60% of the poles had sweeps that were circular or curved in the data set. The reason for this is the noise in the range measurements and lower point density when the distance to the scanner is longer. Also small part of the poles did not have circular cross-section in reality.

### 2.3. Field and Data Reference

Reference data was collected manually in the field. Positions of different kinds of poles and pole-like tree trunks were plotted on a city map. In total 173 targets were recorded in the test field, which included 94 poles and 79 tree trunks. Poles included 43 lamp posts, 37 traffic signs and 14 other poles. Trees that did not have one clear trunk but several branches were not included in the reference. In addition, poles' and trunks' lengths and thicknesses (diameters at breast height) were measured approximately. For the tree trunks, the length was measured from the ground to the end of the pole-like trunk and for the traffic signs from the ground to the bottom of the highest board in the sign. The length was measured with an accuracy of 0.5 m. All reference targets were limited to those that were closer than 30 metres from the scanner route and to those that were longer than one metre.

All poles found in the field were searched for in the collected data set. If a pole was also identified from the range data, it was accepted to a second reference data set called *data reference*. From now on, the original reference collected in the field is referred to as *field reference*. Totally 148 of the 173 field-mapped reference poles were found in the data by the visual inspection. The original field reference and the following data reference based on manual investigation can be used to evaluate the quality of MMS data in pole detection. The data reference can be used to evaluate the performance of the automatic pole detection algorithm and this is the emphasis in this paper. In addition, the field reference is analysed shortly.

## 3. Developed Method

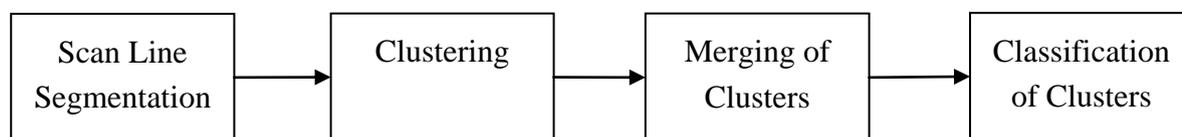
As already mentioned in Section 2.2, slightly more than half of the poles' and tree trunks' sweeps were curved or circular in the data. Consequently, cylinder and trunk detection methods that expect this property would detect only about 60% of the poles in the best case. Data density in VLS is significantly lower compared to TLS (e.g., with the ratio of 1:1,000). In addition, the density of data in this study was much higher along scan lines than in driving direction which resulted in gaps between profiles. Therefore, methods developed for TLS cylinder detection would not be applicable without modifications in this study. For example, many of the TLS methods need quite accurate surface normals for each point. The gaps between profiles make it more difficult or even impossible to calculate surface normals or at least their accuracy would probably not be sufficient. In addition, horizontal slicing of data and calculation of local point distribution are more difficult, especially when the driving speed and thus the data density varies in the data set. The difficulty of applying TLS methods for VLS has also been pointed out by Brenner [1].

A different type of method was developed in this study that exploits the structure of the point cloud, referring to the use of the profile information of the scanner. It is expected that at least three sweeps will be gained from the pole. If a sweep of a pole is found in some profile then one knows that there has to be another sweep either below the current sweep in the next profile or above in the previous

profile or both of them. The algorithm is divided into four phases (Figure 3). First, possible sweeps are extracted from the data. This is done by segmenting each profile separately into point groups that consist of points that are close to each other. Then long groups are removed leaving short point segments that are possible sweeps of poles. In the second phase point groups are clustered resulting in a set of candidate pole clusters. After clustering some poles might be divided into many clusters and they have to be merged. This is done in the third phase of the algorithm. In the last phase, candidate clusters are classified as poles or non-poles. Features used in the classification include shape, length, orientation and point density in the local neighbourhood of the cluster. The values of the parameters of the method used in this study are described at the end of each section.

In addition to points describing real object surfaces, laser scanners, especially phase-based laser systems, produce outliers, *i.e.*, points lying in the air. These are typically removed in the pre-processing of the data [46]. However, no outlier removal was necessary before the pole detection algorithm. First, single points or small point groups lying in the air are removed quite effectively after scan line segmentation and, secondly, there is only a small probability that outlier measurements would form pole like structures.

**Figure 3.** Phases of the pole detection algorithm.

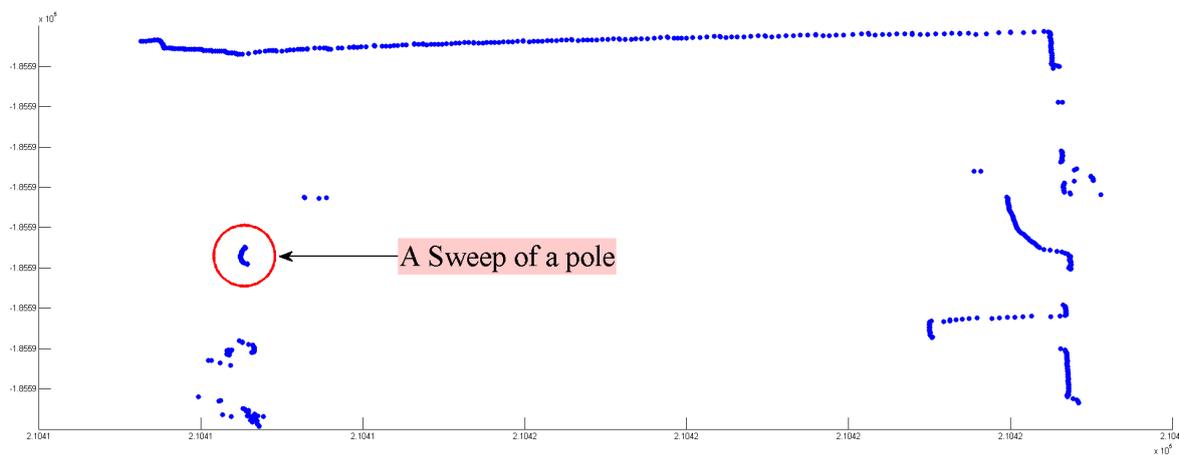


### 3.1. Scan Line Segmentation

In the first phase of the algorithm each scan line is segmented separately. Scan line segmentation is considered, *e.g.*, in [40,49-52]. The method developed in this study includes a modification to previously published methods. The original distance measurements were not available and thus segmentation was done on 3D points. One sweep of a pole consists of few adjacent points on a scan line which have a rather short distance between each other. If a pole is isolated from buildings and vegetation (which is quite a typical situation) one sweep forms an isolated point group on a profile (Figure 4). In other words, there are so-called jump edges on both sides of a sweep on a scan line and the sweep can be segmented by a simple edge detection method.

The edge detection used in this study is based on the distance between adjacent points on the profile. Let  $v_{i,i+1}$  denote this distance, *i.e.*,  $v_{i,i+1}$  is the distance between points with indices  $i$  and  $i + 1$  on the profile. Segmentation is started by initialising an empty group, taking the first point on the profile, *i.e.*, by setting  $i \leftarrow 1$ , and adding it to the group. Then the distance to the second point on the profile ( $v_{i,i+1}$  is calculated. If it is shorter than a maximum allowed distance, point  $i + 1$  is added to the same group with point  $i$ . Otherwise a new empty group is initialised and point  $i + 1$  is added to that. Then the same is repeated for all  $i = 1, \dots, I - 1$  sequentially, where  $I$  is the number of points on the profile. Distance  $v_{i,i+1}$  is the Euclidean distance in three dimensional Cartesian coordinate system.

**Figure 4.** A horizontal projection of a profile with a sweep of a pole circled. The sweep can be isolated from other points by simple scan line segmentation.



The segmentation is repeated for all scan lines and for each segmented point group the information on the profile's index, points' indices on the profile and points' maximum and minimum  $x$ - and  $y$ -values ( $x_{\min}$ ,  $x_{\max}$ ,  $y_{\min}$ ,  $y_{\max}$ ) are recorded. When using a tilted position of the scanner, sweeps of a pole are rather short point groups. The length of the sweep depends on the tilting angle and the width of the pole. Thus after segmentation point groups, which are longer than a maximum allowed length, are removed. The length of a group is the Euclidean distance between the first and last point of the group along the scan line. Point groups that have less than some minimum allowed number of points are also removed. This way single outlier points can be removed. This also lowers the probability of having false positives in the pole detection, *i.e.*, detecting a pole when a pole is not present. In addition, the number of unnecessary groups is lowered which reduces the computational burden of the algorithm.

Only adjacent points on a scan line are compared to each other in segmentation. The drawback of this approach is that missing points in a sweep caused by shadowing or an outlier measurement can split the sweep into many groups. This could be fixed by merging groups that are close to each other or by doing the comparison between two or more adjacent points or by comparing the point to all other points on the profile. However, in the merging of clusters (Section 3.3) the split sweeps can be merged again and comparison to only one adjacent point is sufficient. The advantage of comparing only neighbouring points is the computational efficiency.

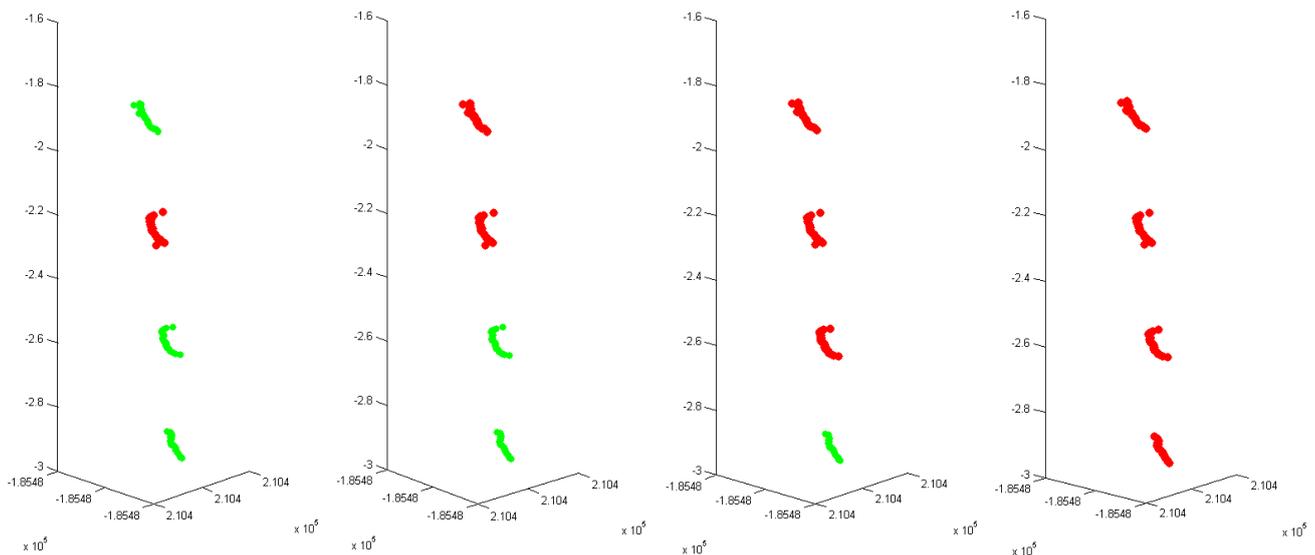
In this study, the maximum distance of two adjacent points on a profile was 10 cm. This was approximately four times longer than the point spacing with 15 Hz mirror frequency and 30 metres distance from the scanner. The maximum length of an accepted point group was one metre and the minimum number of points in a group was five.

### 3.2. Clustering

After scan line segmentation the remaining short point groups are clustered by searching groups that are on top of each other in adjacent profiles. This way the shape of a pole is taken into account and the clusters are good candidates for poles. Clustering is done in the horizontal plane, *i.e.*, in two-dimensional space.

Firstly, all point groups are saved in a list and one group is selected from some profile with, say, index  $j$  as a seed group. If it is part of a pole then there should be another group on top of it in the previous profile or below it in the next profile or both of them. First the previous profiles are checked. The seed group is compared to all groups in the previous profile with index  $j-1$ . If a group is found above the seed group it is added to the same cluster with the seed group. The comparison of two groups is described below. Next the added group in profile  $j-1$  is compared again to all groups in the previous profile (index  $j-2$ ) and if a new group is found it is added to the cluster. Clustering continues this way until no group is found from a previous profile. Next, the search starts again from the seed group to the next profiles (Figure 5). The procedure is identical to that in previous profiles. After both directions have been checked, the whole cluster is saved and point groups belonging to it are removed from the list of all groups. Then a new seed group is selected from the list of all groups and clustering is repeated. The whole procedure is continued until all groups are clustered and the list of all groups is empty.

**Figure 5.** Clustering of the point groups. Red colour indicates that a group is already added to the cluster. At each step (from left to right) the highest (lowest) group of the cluster is compared to the groups of the previous (next) profile. If a group in approximately same position in the horizontal plane is found, it is added to the cluster.



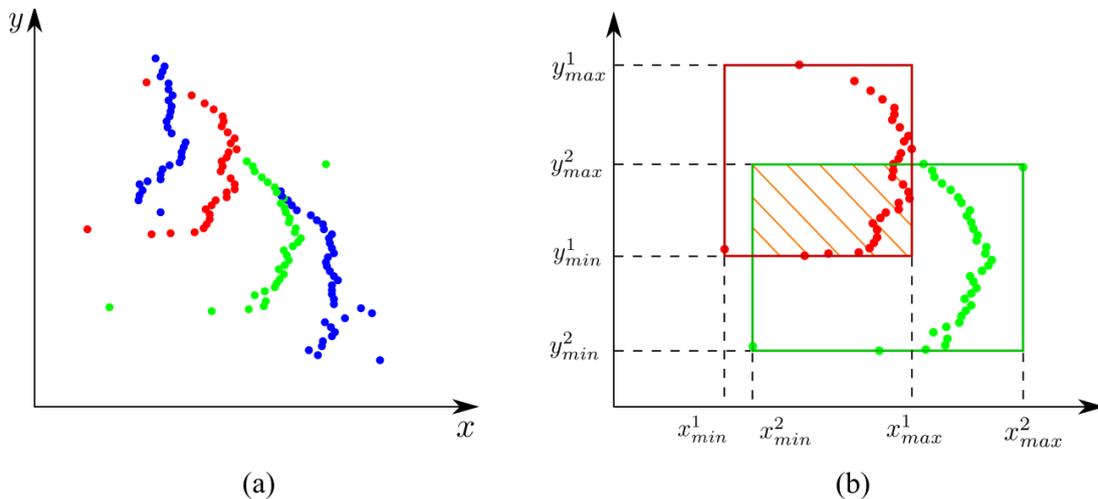
A point group that is already added to a cluster is compared to all point groups in the previous (next) profile. A group that is on top of (below) it is searched for and the comparison is done in the horizontal plane (Figure 6a). Here the seed group of a cluster is taken as an example to illustrate the idea. The procedure is identical for all groups that are later added to the cluster. Let  $g_1$  be the seed group of a cluster that is compared to another group  $g_2$ , which is in the previous profile. Let the two groups have maximum and minimum  $x$  and  $y$  values  $(x_{\min}^{(1)}, x_{\max}^{(1)}, y_{\min}^{(1)}, y_{\max}^{(1)})$  and  $(x_{\min}^{(2)}, x_{\max}^{(2)}, y_{\min}^{(2)}, y_{\max}^{(2)})$ , respectively. Rectangles are placed around both groups, one for each group (Figure 6b). The sides of the rectangles are parallel to the coordinate axes. The upper right corners of the rectangles are in  $(x_{\max}^{(k)}, y_{\max}^{(k)})$  and the lower left corners in  $(x_{\min}^{(k)}, y_{\min}^{(k)})$ , where  $k = 1, 2$ .  $g_2$  is added to the cluster if the rectangles around the groups overlap. As mentioned in Section 2.2, poles are not necessarily accurately

vertical and thus sweeps are not always accurately on top of each other. That is why a positive constant, denoted by  $\Gamma_{\text{rect}}$ , is added to the side lengths of the rectangles in the  $x$ - and  $y$ -directions. Mathematically,  $g_2$  is added to the cluster if

$$\begin{aligned} & \left(x_{\text{max}}^{(2)} \geq x_{\text{min}}^{(1)} - \Gamma_{\text{rect}}\right) \wedge \left(x_{\text{min}}^{(2)} \leq x_{\text{max}}^{(1)} + \Gamma_{\text{rect}}\right) \wedge \\ & \left(y_{\text{max}}^{(2)} \geq y_{\text{min}}^{(1)} - \Gamma_{\text{rect}}\right) \wedge \left(y_{\text{min}}^{(2)} \leq y_{\text{max}}^{(1)} + \Gamma_{\text{rect}}\right). \end{aligned} \tag{1}$$

Sometimes many groups are found whose rectangles overlap with the rectangle of  $g_1$ . This is caused, e.g., by small groups of tree leaves that are close to each other. Sweeps of a pole and tree trunk are sometimes cut into many pieces due to shadowing or roughness of a trunk which causes a similar effect. If many groups are found only one of them is selected randomly and added to the cluster. The rest are not discarded but clustered later. This causes a problem that part of a sweep might be missing in a pole cluster. However, the next phase of the algorithm where the clusters are merged compensates this problem to some extent.

**Figure 6.** (a) A pole cluster in MMS data projected to the horizontal plane. Different sweeps are in different colour. (b) Comparison of green and red groups in (a). The groups belong to a same cluster if there exist an overlapping of the red and green rectangles (ruled area). The extra side length  $\Gamma_{\text{rect}}$  is not taken into account in the figure.



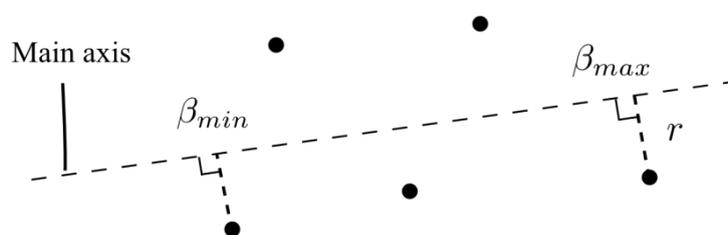
Only adjacent profiles are compared in the clustering. The drawback of this approach is that sometimes there are missing sweeps due to shadowing and thus the pole is divided into many clusters. This problem is partially fixed in the next phase of the algorithm where clusters are merged. If the comparison was made to more than one adjacent profile, bigger  $\Gamma_{\text{rect}}$  would have to be used. In this study,  $\Gamma_{\text{rect}}$  was 0.05 m. Tests with greater  $\Gamma_{\text{rect}}$  have given worse performance than with the current value and thus comparison to only one adjacent profile was preferred.

### 3.3. Merging of Clusters

As discussed earlier, some poles might be divided into several clusters. There are two reasons for that. First, some sweeps might be missing due to shadowing from other objects between the scanner and the pole. Second, some sweeps might be cut into many pieces. In addition to shadowing, outlier

measurements and, for example, roughness of a tree trunk surface may be reasons for this. Thus the clusters need to be merged. Not all candidates that were found in clustering are pole-like or vertically oriented. The objective in the merging phase of the algorithm is to find all clusters that are pole-like and vertically oriented, compare them to all other clusters and merge them with all clusters that seem to belong to the same pole. In the following, pole-like vertically oriented clusters are called *base clusters*. Shape and orientation of a cluster is inspected by performing the PCA to the points of the cluster. The cluster is defined as pole-like if the ratio of the variance in the first principal direction of the cluster to the variance in the second principal direction is more than a threshold. Furthermore, the main axis of the cluster is the line that goes through the middle point of the cluster and is parallel to the direction of the first PC (Figure 7). To be vertically oriented, cluster's orientation has to be close enough to the  $z$ -axis, *i.e.*, the angle between the  $z$ -axis and cluster's main axis has to be small enough.

**Figure 7.** Orientation (main axis, long dashed line) and point's (black dots) perpendicular distance from the main axis ( $r$ ) illustrated in 2D.  $\beta_{\max}$  and  $\beta_{\min}$  are the maximum and minimum projections on the main axis (see Section 3.4).



If a cluster belongs to the same pole with the base cluster, cluster's points' distance from the base cluster's main axis should be similar to the distance of the points in the base cluster. Therefore, for each point in the cluster the perpendicular distance from the point to the main axis of the base cluster is calculated ( $r$  in Figure 7). Let  $\bar{r}$  denote the mean perpendicular distance of the points in the cluster and  $\bar{r}_{\text{base}}$  the same for the base cluster. If  $\bar{r}$  is similar to  $\bar{r}_{\text{base}}$ , the cluster might belong to the same pole as the base cluster. There should also not be too large separation between the two clusters, which is inspected by calculating the distance between the upper clusters lower end and lower clusters upper end. Finally, the clusters are merged if the following conditions are met:

- (i)  $\bar{r} \leq \bar{r}_{\text{base}} + \Gamma_{\bar{r}}$ , where  $\Gamma_{\bar{r}}$  is a small positive constant.
- (ii) The distance between the two clusters is not too long.

Base clusters are chosen one at a time in the order of their length so that the longest cluster is chosen first and the shortest last. Each of the base clusters are compared to all shorter clusters, including all shorter base clusters. All clusters that fulfil the above conditions (i) and (ii) with the base cluster are merged with it. After that, the merged clusters are not compared to any other base cluster any more. This applies also to shorter base clusters that have been merged with some other base cluster. To reduce the computational burden, for each selected base cluster only the closest clusters from the remaining shorter clusters are chosen for comparison. In this study, the minimum ratio of the variances in the PCA was set to  $4^2$ .  $\Gamma_{\bar{r}}$  was 0.1 m, the maximum distance between clusters 2.5 metres and the maximum allowed angle between the axis of the cluster and the  $z$ -axis 30 degrees.

### 3.4. Classification of Candidate Clusters

A large number of candidate clusters are found, e.g., in building walls due to shadowing. In the final phase of the algorithm candidates are classified as poles or non-poles. To be accepted as a pole, the candidate has to fulfil the following requirements:

- (a) The cluster's length along its main axis has to be high enough.
- (b) The cluster has to consist of minimum three sweeps.
- (c) The cluster has to have a pole-like shape.
- (d) The main axis of the cluster should be close to vertical.
- (e) There should not be too many points around the cluster in its close neighbourhood.

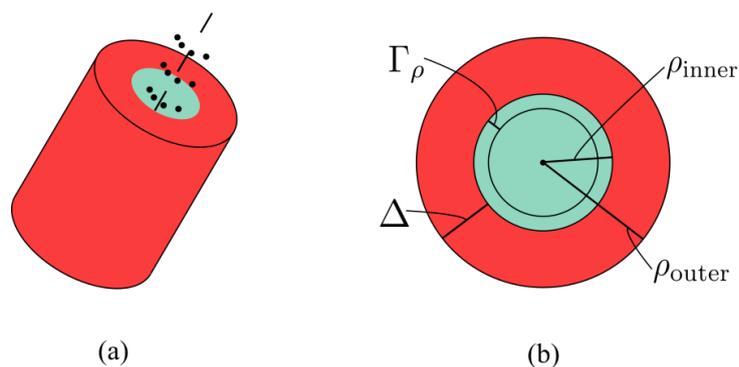
As the condition (a) says, the aim is to find poles that are longer than some minimum length and therefore too short clusters are removed before classification. Calculation of the length of the cluster is started by projecting all points of the cluster to its main axis. The length of the cluster is given by the difference between the largest and smallest projection ( $\beta_{\max}$  and  $\beta_{\min}$  in Figure 7). Due to the gaps between the profiles (37 cm on average in the data set) the length of a pole calculated straight from the projections is biased and too low. The maximum error is two times the gap width and on average there is one gap width missing (half a gap width from both ends). Therefore a gap width is added to the length of the cluster which gives a better approximation of the length of the pole on average. Because the gaps between the profiles are not constant they are approximated separately for each cluster by calculating the average distance between adjacent sweeps in the cluster. In addition, it is required that the cluster contains at least three sweeps. This way the number of false positives can be lowered because clusters where two sweeps are accidentally on top of each other are discarded. The drawback of this restriction is that some poles which are hit by only two sweeps are discarded. The conditions (c) and (d) are checked as in the selection of base clusters in the merging phase using the same threshold values for ratio of variances and angle between the main axis of the cluster and the  $z$ -axis.

The condition (e) is aimed at removing clusters that are accidentally formed from, e.g., building walls and branches or leaves of trees. In this case there are typically other points from the wall or other branches around the cluster. If these are found it is possible to classify the cluster as non-pole. Classification presented in this paper is somewhat similar to the method developed by Brenner [1] where data is analysed in cylindrical stacks. The condition is checked by placing two cylinders on the cluster (Figures 8a and 8b). The main axes of the cylinders coincide with the main axis of the cluster (cluster and cylinders are coaxial). The upper and lower ends of the cylinders are the same as upper and lower end of the cluster. Inner cylinder approximates the pole that the cluster possibly presents. Its radius, denoted by  $\rho_{\text{inner}}$ , is chosen so that there would only be points which belong to the expected pole inside the cylinder. In addition, all points of the cluster should fall inside the cylinder. A good approximation of the radius is the maximum perpendicular distance of the cluster's points from the main axis, denoted by  $r_{\text{max}}$ . In addition, a small extra length  $\Gamma_{\rho}$  is added to the radius and hence  $\rho_{\text{inner}} = r_{\text{max}} + \Gamma_{\rho}$ . The outer cylinder's radius, denoted by  $\rho_{\text{outer}}$ , is somewhat larger and is given by  $\rho_{\text{outer}} = \rho_{\text{inner}} + \Delta$ , where  $\Delta$  is the difference between the radii. If the cluster is a pole, the number of points inside the outer cylinder should not be much higher than the number of points inside the inner cylinder. More precisely, if the number of points inside the inner cylinder is greater than  $P\%$  of the

number of points inside the outer cylinder, the cluster fulfils the condition (e). The points inside the cylinders are searched for from the original point cloud.

The checking of the neighbourhood using cylinders (from now on *the cylinder method*) is not completely straightforward. Some poles might also include some other parts and only part of the cluster is pole-like. For example, traffic signs contain boards (Figure 2b). When the cylinder method is applied for a sign as a whole the board might cause the rejection of the pole. Consequently, a more detailed check of the cluster is needed and the cluster is inspected in pieces. First, the two lowest sweeps are chosen and it is checked whether the piece is pole-like and vertical. If the conditions (c) and (d) are not met, the next lowest sweep is added to the piece. Sweeps are added to the piece until it is pole-like and vertical. Detection rate is improved if it is also insisted that the piece has to be longer than a minimum length that is lower than the minimum length of the pole. If a valid piece is found the cylinder method is applied to it and the piece is either accepted or ignored. The same procedure is repeated, *i.e.*, new pole-like vertical pieces are searched for and checked with the cylinder method starting from each sweep of the cluster. Finally, all accepted pieces are merged into one point cluster. If the cluster satisfies the conditions (a) and (b), it is accepted as a pole. Otherwise the cluster is rejected.

**Figure 8.** The cylinder method. (a) Two coaxial cylinders (red and light green) are placed on the cluster so that the inner cylinder approximates the expected pole. The cluster may be accepted as a pole only if the red volume around the cluster is almost empty of points. For illustration purposes the lengths of the cylinders are not equal to the length of the cluster. (b) The radii of the cylinders.



Part of the road surface can be removed from the original data set before the cylinder method to speed up the algorithm. It is easy to remove the centre of the road because it is the middle part of each profile. Before the cylinder method, only the closest points that are inside a cube that is around the cluster are selected. This way one does not have to use the whole data set when calculating distances from the main axis. The minimum length of the pole was the same as in the reference data (1 metre). In the cylinder method, the minimum length of the piece was 0.5 m,  $\Gamma_\rho$  was 0.1 m,  $\Delta$  was 0.35 m, and  $P$  was 95%.

#### 4. Results and Discussion

In this section the results of the testing of the method are presented and analysed. At the end of the section some issues concerning the algorithm are discussed and future work is anticipated. Analysis of

the results is divided into two parts: the first part discusses the data reference and the second part discusses the field reference.

#### 4.1. Data Reference

Out of the 148 data reference poles, the developed algorithm detected 115 poles and 33 poles were missed. Algorithm detected totally 142 targets and thus there were 27 false positives, *i.e.*, clusters that did not belong to the reference data set. Thus, the completeness of the detection was 77.7%, correctness 81.0% and mean accuracy 79.3% (Table 3). Mean accuracy (*MA*) was defined as the probability that a randomly chosen automatically detected target was correctly detected and that a randomly chosen reference target was detected by the algorithm [53]. The formula for the mean accuracy was [53]:

$$MA = \frac{2 \times \text{number of correctly detected targets}}{\text{number of reference targets} + \text{number of all detected targets}} \cdot 100\% \quad (2)$$

When only targets (detected and reference) closer than 12.5 m from the scanner route were considered in the analysis the completeness was 5.8% and correctness 5.5% higher than with the original data reference (Table 3). 12.5 m distance was chosen because most of the reference poles and tree trunks (approximately three quarters) were closer than this from the scanner route. The reason for the better detection rate closer to the scanner is higher point density and lesser effect from shadowing from other objects. The difference in the detection rates would probably be bigger if the poles were more uniformly distributed in all distances from the scanner route.

**Table 3.** Results of the automatic pole detection using the data reference. *d* is the maximum allowed distance from the targets to the scanner route. In the middle column are the results for all targets that are closer than 30 m from the scanner route. In the right column only targets that are closer than 12.5 m from the scanner route are taken into account.

Statistical Number	Data Reference	Data Reference
	<i>d</i> < 30 m	<i>d</i> < 12.5 m
Completeness (%)	77.7	83.5
Correctness (%)	81.0	86.5
Mean Accuracy (%)	79.3	85.0

Fourteen of the missed reference poles had vegetation around them, the tree trunk was partially inside branches or there were some scattered points around the poles. These poles were discarded by the cylinder method because there were too many points in the outer cylinder. Four missed poles had sweeps that contained less than five points and were thus discarded after scan line segmentation. Five missed tree trunks were behind parked cars which meant that the trunk was only partially visible in the data. This problem can be seen in Figure 2a where there are “black shadows” between parked cars and buildings on the right side of the road. Three missed poles were so oblique or curved that the sweeps were not clustered and no candidate cluster was formed. The rest seven reference poles were missed, for example, because there were too many sweeps missing in the pole due to shadowing, the pole was

so short that it consisted of only two sweeps in the data or the pole was a traffic sign which had two boards so that the pole-like parts were too short. The application of the cylinder method loses sometimes the highest sweep of the pole. This caused the rejection of at least one reference pole which consisted of only three sweeps.

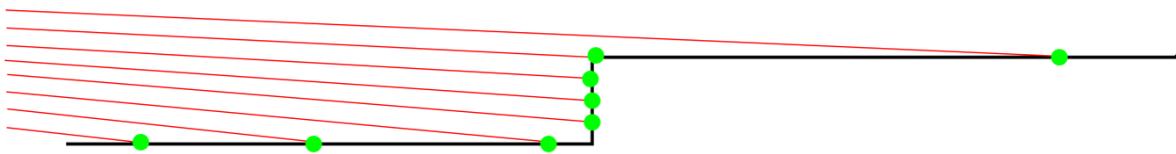
Because the parameter  $P$  was as high as 95% in the cylinder method, only a few points around the pole caused a rejection of the pole. However, if the percentage was lowered the correctness of the classification would be poorer. If the minimum number of points in one group in scan line segmentation was set to less than five no significant improvements were achieved in mean accuracy. The computation time of the algorithm almost doubled when smaller groups were accepted and thus minimum of five points was preferred. The problem of parked cars could be diminished by placing the scanner higher in the measuring vehicle or, if possible, simply by performing the data collection during a time when there are only few parked cars. The oblique poles and curved tree trunks could be detected possibly by applying some different kind of clustering method.

14 of the 27 falsely detected targets were pillars in the buildings. Before collection of field reference it was decided that pillars would not be included in it. However, there is no feature in the algorithm that would separate pillars from other poles and in this aspect the definition of the reference was not completely successful. If pillars were included in the reference, the correctness of the detection would have been more than 90%. However, in this case the completeness value would also have been different.

Seven of the rest 13 false positives were found from different wall structures in the buildings. The reasons for these were scanning geometry and shadowing. Surfaces that are nearly parallel to the laser rays will have a rather low point density. Therefore, when the angles between a building wall and the laser rays are small, only a very small amount of measurements will be achieved from the wall. On the contrary, planes perpendicular to the laser ray have a much higher point density. Thus narrow corners in the aforementioned walls can produce pole-like point clusters into the data (Figure 9). Also parts of the walls that are between windows, doorways and edges of buildings can cause narrow point clusters that look like poles. The edges of a wall can be problematic because there may not be points on the other side of the edge at all.

Two of the rest falsely detected targets were branches of a tree that did not have one clear trunk but several branches. One false positive was caused by an error in the georeferencing of the data that caused that a pole was divided into two parts which were not coaxial and one of them was considered falsely detected. One false positive was a traffic sign that was in the field reference but could not be identified as a pole in the manual check of the data. One incorrect target was a pole whose length was less than one metre in the field and therefore it was not included in the reference. The approximation of the length calculated by the algorithm was in this case too long and therefore the pole was not discarded in the method. The last falsely detected target may have been a very small tree which was not in the reference. The number of falsely detected poles from the buildings could be lowered by also using the data collected from the opposite driving direction. In some part of the modelling process the buildings are also extracted. After this it may be possible to remove the falsely detected poles from the building structures.

**Figure 9.** Density of the laser measurements (green dots) on a wall (black line) that is nearly parallel to the laser rays (red lines) and that has a narrow corner.



Concerning the subclasses of poles, lamp posts were the easiest to detect having approximately 17–20% higher detection rate than traffic signs or tree trunks (table 4). The most significant factor causing the differences in the detection rates was probably the length of the pole. Almost all of the lamp posts were more than three metres long while this was the case for less than half of the other targets.

**Table 4.** Detection rates for different subclasses of poles in the automatic detection.

Subclasses	Detection Rates (%)	
	Data Reference	Field Reference
Traffic Signs	73.3	59.0
Lamp posts	93.0	93.0
Tree Trunks	76.1	64.6
Targets with Length $\geq 2.5$ m	–	81.6
Targets Thicker than 10 cm	–	78.8

#### 4.2. Field Reference

85.5% of poles found in the field were also found in the data when it was checked manually. All lamp posts, 76.9% of traffic signs and 84.8% of trees could be identified. The reason for not finding a pole in the data was the shadowing effect from other objects between the pole and the scanner. The detection rate could possibly be improved to some extent if data from two opposite driving directions was used. Completeness, correctness and mean accuracy of the automatic detection were 67.1%, 81.7% and 73.7%, respectively (Table 5). As with the data reference, lamp posts were the easiest to detect from the subclasses (Table 4). When only targets that were closer than 12.5 m from the scanner route were taken into account completeness was 9.3%, correctness 5.7%, and mean accuracy 7.9% higher than with the original reference (Table 5).

Detection rate for the poles whose length was 2.5 m or higher was 14.6% higher when compared to the original completeness value (Tables 4 and 5). Correspondingly, poles which were thicker than 10 cm had 11.8% higher detection rate. Approximately half of the reference targets were 2.5 m long or longer and about 60 % were thicker than 10 cm. The higher detection rates for thicker and longer poles are due to the fact that they have larger area and thus there is higher probability that the laser will hit them. However, these restrictions would rule out a large part of poles in road environments. For example, all poles in traffic signs were 6 cm thick and more than half of them also less than 2.5 m long.

**Table 5.** The corresponding results as in table 3 for the field reference.

Statistical Number	Field Reference	Field Reference
	$d < 30$ m	$d < 12.5$ m
Completeness (%)	67.1	76.4
Correctness (%)	81.7	87.4
Mean Accuracy (%)	73.7	81.5

#### 4.3. Validation of the Method

In the results presented so far, the whole data set was used for the development of the method, and therefore the results may be too optimistic. Another test was conducted in which the data set was divided into two parts. Profiles 600–1,000 and the corresponding reference targets formed the training set and profiles 1–599 with the corresponding reference targets the validation set. The number of reference targets was approximately same in both sets. The training set was used for tuning the parameters of the method and validation set was used to test the performance of the method.

The completeness of the detection was less than one percent lower with the validation set when compared to the completeness with the training set. Correctness values were not comparable because false positives from the buildings (pillars and narrow corners) were unevenly divided between the training and validation sets.

Because the whole data set (training and validation together) had been used to develop the algorithm, the validation set was not completely independent and the validation results may be optimistic. However, based on the results, the parameters of the method do not seem to be highly sensitive and overfitted to the used data set. Therefore, the results presented for the whole data set (Sections 4.1 and 4.2) should give a good estimate of the performance of the method also with new unseen data.

#### 4.4. Applicability of the Method in Other MMS

The method described in this paper is applicable to MMS whose data consists of profiles and where the scanning plane is tilted which is the case, e.g., for the StreetMapper 360 and Lynx systems. In addition, PRF should be large enough in relation to mirror frequency so that several points will hit even a narrow pole in one profile. The parameters of the method depend on and need to be adjusted according to the characteristics of the used MMS. These include, e.g., tilting angle of the scanner, driving speed, mirror frequency and PRF. The information on the relative positions of profiles and points on each profile are not necessary and with some modifications the method is applicable even if this information is not available.

#### 4.5. Future Work

In the future, the method will be developed further and tested with larger and different kinds of data sets. These will include combined data from two opposite driving directions in the same area, data collected with different MMS and with different driving speeds and scanner frequencies. Use of combined data sets from different scans of the same area may improve the detection rate of the

method. It is also expected that the density and quality of VLS data are increased in the future which could also result in higher accuracies. Separation of tree trunks from other poles will also be a topic of research.

The detection rate of the poles might be increased by developing the method further. Poles and trunks which are visible in the data but have substantially many points around them (e.g., due to branches of trees) are not classified as poles by the current algorithm and need different methods. Also detection of tilted poles and poles that have many missing sweeps need some further research. Sometimes the highest sweep of the pole is lost when the cylinder method is applied and the method needs to be improved. The Roamer system includes also cameras which could also be used in the detection of the traffic signs [54] which had the poorest visibility in the laser data. Due to the huge amounts of data produced by VLS systems, the computational efficiency of the method needs to be investigated.

## 5. Conclusions

Accurate road environment information is needed in numerous applications, such as road maintenance, 3D city and noise modelling, location-based services and driver assistance systems. Vehicle-based laser scanning (VLS) supplements conventional laser-based mapping techniques; it produces more accurate and denser data than ALS and is much more efficient than TLS when large areas are mapped. Poles can be considered as an important part of road infrastructure; they stand out from the environment and can be used as reference targets in different applications. Examples of pole-like objects include utility poles, such as traffic signs and lamp posts, and tree trunks. Because of differences in point density the data processing methods for ALS and TLS are usually not applicable to VLS as such and new methods have to be developed.

The aim of the study was to develop automatic methods for the extraction of pole-like objects in road environments based on VLS data. Because of large amounts of data produced by VLS the data processing methods need to be automatic. Up to this date, a few methods have been developed for pole detection in VLS but a thorough experimental testing has not been reported for any of them. One of the main objectives of the study was to analyse the statistical accuracy of the developed method.

The developed pole extraction method was tested in a suburban area and an approximately 450 m long straight stretch of road was mapped with the Roamer MMS. The road was driven in one direction with an average driving speed of approximately 20 km/h. For the range measurements, the phase-based Faro LS 880HE80 laser scanner was used with 120 kHz PRF, 320 degrees FOV in the scanning plane, 15 Hz mirror frequency and about 50 metres maximum measurement distance. The scanning plane was tilted 135 degrees backwards with respect to vehicle motion which made it possible to achieve multiple sweeps from the poles from adjacent profiles. The whole data set consisted of about seven million 3D points. Reference data was collected manually from the test field and only targets that were closer than 30 metres from the scanner route were included. The main classes of the targets were lamp posts, traffic signs and tree trunks.

The quality of VLS data proved to be good in pole detection. 85.5% of the poles found in the field were manually identified from the collected laser data. These formed another reference called data reference which was used for the evaluation of the developed automatic algorithm. From data

reference poles, the developed algorithm found 77.7%. Correctness of the classification was 81.0%. Poles inside vegetation, tree trunks inside branches and behind parked cars, poles that had some scattered points around them and too oblique poles were the most difficult to find with the developed algorithm. More than half of the falsely detected targets were pillars which were intentionally left out from the reference. More than one quarter of the false positives came from building structures because of scanning geometry and occlusion. It might be possible to remove these in later stages of the modelling process when buildings are extracted. Traffic signs were the most difficult and lamp posts the easiest to detect both with the automatic method and in the manual investigation of the data. As could be expected the completeness was higher for poles and trees that were closer to the road, had larger diameter or were longer. Generalisation properties of the method for new unseen data seem promising because the parameters of the method did not seem to be overfitted to the used data set.

With some modifications the method should be applicable in different laser-based MMS, also in the most prominent commercial systems. The accuracy of the detection may possibly be increased by developing the method further. Improved accuracy may also be achieved by combining data from several scans of the same area and due to the increased density and quality of VLS data in the future. The computational efficiency of the developed method needs to be investigated and the method will be tested with different and larger data sets.

## Acknowledgements

This study was part of the “Development of Automatic, Detailed 3D Model Algorithms for Forests and Built Environment—Built” and 3D-NAVI-EXPO projects, both supported by the Finnish Funding Agency for Technology and Innovation (Tekes). The authors wish to thank X. Liang for the discussion about the development and testing of the method and L. Matikainen for the help with the terminology and evaluation of the method. The authors are also grateful for the reviewers for several suggestions to improve the manuscript.

## References

1. Brenner, C. Extraction of Features from Mobile Laser Scanning Data for Future Driver Assistance Systems. In *Proceedings of the 12th Agile Conference*, Hannover, Germany, 2009.
2. Chen, Y.-Z.; Zhao, H.-J.; Shibasaki, R. A Mobile System Combining Laser Scanners and Cameras for Urban Spatial Objects Extraction. In *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, Hong Kong, China, 2007.
3. Manandhar, D.; Shibasaki, R. Feature Extraction from Range Data. In *Proceedings of the 22nd Asian Conference on Remote Sensing*, Singapore, 2001.
4. Shi, Y.; Shibasaki, R.; Shi, Z. Towards Automatic Road Mapping by Fusing Vehicle-Borne Multi-Sensor Data. *Int. Arch. Photogramm. Remote Sens.* **2008**, XXXVII–B5, 867–872.
5. Doubek, P.; Perd'och, M.; Matas, J.; Šochman, J. Mobile Mapping of Vertical Traffic Infrastructure. In *Proceedings of the 13th Computer Vision Winter Workshop 2008*, Ljubljana, Slovenia, 2008.
6. Li, R.; Wang, W.; Tseng, H.Z. Detection and Location of Objects from Mobile Mapping Image Sequences by Hopfield Neural Networks. *Photogramm. Eng. Remote Sens.* **1999**, 65, 1199–1205.

7. Hyypä J.; Hyypä H.; Yu, X.; Kaartinen, H.; Kukko, A.; Holopainen, M. Forest Inventory Using Small-Footprint Airborne LiDAR. In *Topographic Laser Ranging and Scanning, Principles and Processing*, Shan, J., Toth, C.K., Eds.; CRC Press: Boca Raton, FL, USA, 2009.
8. Sithole, G.; Vosselman, G. Experimental Comparison of Filter Algorithms for Bare-Earth Extraction from Airborne Laser Scanning Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2004**, *59*, 85-101.
9. Hyypä J.; Hyypä H.; Litkey, P.; Yu, X.; Haggrén, H.; Rönholm, P.; Pyysalo, U.; Pitkänen, J.; Maltamo, M. Algorithms and Methods of Airborne Laser Scanning for Forest Measurements. *Int. Arch. Photogramm. Remote Sens.* **2004**, *XXXVI-8/W2*, 82-89.
10. Brenner, C. Building Reconstruction from Images and Laser Scanning. *Int. J. Appl. Earth Obs. Geoinf.* **2005**, *6*, 187-198.
11. Maas, H.G.; Vosselman, G. Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data. *ISPRS J. Photogramm. Remote Sens.* **1999**, *54*, 153-163.
12. Ahokas, E.; Kaartinen, H.; Matikainen, L.; Hyypä J.; Hyypä H. Accuracy of High-Pulse-Rate Laser Scanners for Digital Target Models. In *Observing Our Environment from Space: New Solutions for a New Millennium: Proceedings of the 21st EARSeL Symposium, Paris, France, 14-16 May 2001*; Balkema: Leiden, the Netherlands, 2002; pp. 175-178.
13. Stamos, I.; Allen, P. 3-D Model Construction Using Range and Image Data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC, USA, 2000; pp. 1531-1536.
14. Boulaassal, H.; Landes, T.; Grussenmeyer, P.; Tarsha-Kurdi, F. Automatic Segmentation of Building Façades Using Terrestrial Laser Data. *Int. Arch. Photogramm. Remote Sens.* **2007**, *XXXVI-3/W52*, 65-70.
15. Pu, S.; Vosselman, G. Extracting Windows from Terrestrial Laser Scanning. *Int. Arch. Photogramm. Remote Sens.* **2007**, *XXXVI-3/W52*, 320-325.
16. Gorte, B.; Pfeifer, N. Structuring Laser-Scanned Trees Using 3D Mathematical Morphology. *Int. Arch. Photogramm. Remote Sens.* **2004**, *XXXV-B5*, 929-933.
17. Pfeifer, N.; Gorte, B.; Winterhalder, D. Automatic Reconstruction of Single Trees from Terrestrial Laser Scanner Data. *Int. Arch. Photogramm. Remote Sens.* **2004**, *XXXV-B5*, 114-119.
18. El-Sheimy, N. An Overview of Mobile Mapping Systems. In *Proceedings of the FIG Working Week 2005 and GSDI-8*, Cairo, Egypt, 2005; pp. 16-21.
19. Jaakkola, A.; Hyypä J.; Hyypä H.; Kukko, A. Retrieval Algorithms for Road Surface Modelling Using Laser-Based Mobile Mapping. *Sensors* **2008**, *8*, 5238-5249.
20. Goulette, F.; Nashashibi, F.; Abuhadrous, I.; Ammoun, S.; Laurgeau, C. An Integrated On-board Laser Range Sensing System for On-the-Way City and Road Modelling. In *Proceedings of ISPRS Commission I Symposium*, Paris, France, 2006.
21. Früh, C.; Jain, S.; Zakhor, A. Data Processing Algorithms for Generating Textured 3D Building Façade Meshes from Laser Scans and Camera Images. *Int. J. Comput. Vis.* **2005**, *61*, 159-184.
22. Yu, S.; Sukumar, S.; Koschan, A.; Page, D.; Abidi, M. 3D Reconstruction of Road Surfaces Using an Integrated Multi-Sensory Approach. *Opt. Lasers Eng.* **2007**, *45*, 808-818.
23. Zhao, H.; Shibasaki, R. Reconstructing a Textured CAD Model of an Urban Environment Using Vehicle-Borne Laser Range Scanners and Line Cameras. *Mach. Vis. Appl.* **2003**, *14*, 35-41.

24. Petrie, G.; Toth, C.K. Terrestrial Laser Scanners. In *Topographic Laser Ranging and Scanning, Principles and Processing*, Shan, J., Toth, C.K., Eds.; CRC Press: Boca Raton, FL, USA, 2009.
25. Ellum, C.; El-Sheimy, N. Land-Based Mobile Mapping Systems. *Photogramm. Eng. Remote Sens.* **2002**, *68*, 13-17.
26. Tao, C. Mobile Mapping Technology for Road Network Data Acquisition. *J. Geospat. Eng.* **2000**, *2*, 1-13.
27. Newman, T.S.; Flynn, P.J.; Jain, A.K. Model-Based Classification of Quadric Surfaces. *CVGIP: Image Underst.* **1993**, *58*, 235-249.
28. Marshall, D.; Lukacs, G.; Martin, R. Robust Segmentation of Primitives from Range Data in the Presence of Geometric Degeneracy. *IEEE Trans. Pattern. Anal. Mach. Intell.* **2001**, *23*, 304-314.
29. Lukács, G.; Martin, R.; Marshall, D. Faithful Least-Squares Fitting of Spheres, Cylinders, Cones and Tori for Reliable Segmentation. In *Proceedings of the Fifth European Conference on Computer Vision*, Freiburg, Germany, 1998.
30. Rabbani, T.; van den Heuvel, F. Efficient Hough Transform for Automatic Detection of Cylinders in Point Clouds. *Int. Arch. Photogramm. Remote Sens.* **2005**, *XXXVI-3/W19*, 60-65.
31. Bolles, R.; Fischler, M. A Ransac-Based Approach to Model Fitting and Its Application to Finding Cylinders in Range Data. In *Proceedings of the 7th IJCAI*, Vancouver, Canada, 1981; pp. 637-643.
32. Chaperon, T.; Goulette, F. Extracting Cylinders in Full 3D Data Using a Random Sampling Method and the Gaussian Image. In *Proceedings of the Vision Modeling and Visualization Conference*, University of Stuttgart, Germany, 2001; pp. 35-42.
33. Schnabel, R.; Wahl, R.; Klein, R. Efficient Ransac for Point-Cloud Shape Detection. *Comput. Graph. Forum* **2007**, *26*, 214-226.
34. Luo, D.; Wang, Y. Rapid Extracting Pillars by Slicing Point Clouds. *Int. Arch. Photogramm. Remote Sens.* **2008**, *XXXVII-B3b*, 215-218.
35. Forsman, P. *Three-Dimensional Localization and Mapping of Static Environments by Means of Mobile Perception*. PhD Thesis, Helsinki University of Technology, Espoo, Finland, 2001.
36. Aschoff, T.; Spiecker, H. Algorithms for the Automatic Detection of Trees in Laser Scanner Data. *Int. Arch. Photogramm. Remote Sens.* **2004**, *XXXVI-8/W2*, 71-75.
37. Bienert, A.; Maas, H.-G.; Scheller, S. Analysis of the Information Content of Terrestrial Laserscanner 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.
38. Bienert, A.; Scheller, S.; Keane, E.; Mohan, F.; Nugent, C. Tree Detection and Diameter Estimations by Analysis of Forest Terrestrial Laserscanner Point Clouds. *Int. Arch. Photogramm. Remote Sens.* **2007**, *XXXVI-3/W52*, 50-55.
39. Király, G.; Brolly, G. Tree Height Estimation Methods for Terrestrial Laser Scanning in a Forest Reserve. *Int. Arch. Photogramm. Remote Sens.* **2007**, *XXXVI-3/W52*, 211-215.
40. Liang, X.; Litkey, P.; Hyypä, J.; Kukko, A.; Kaartinen, H.; Holopainen, M. Plot-Level Trunk Detection and Reconstruction Using One-Scan-Mode Terrestrial Laser Scanning Data. In *Proceedings of the International Workshop on Earth Observation and Remote Sensing Applications*, Beijing, China, 2008; pp. 1-5.

41. Litkey, P.; Liang, X.; Kaartinen, H.; Hyyppä J.; Kukko, A.; Holopainen, M. Single-Scan TLS Methods for Forest Parameter Retrieval. In *Proceedings of the SilviLaser 2008*, Edinburgh, UK, 2008; pp. 295-304.
42. Haala, N.; Reulke, R.; Thies, M.; Aschoff, T. Combination of Terrestrial Laser Scanning with High Resolution Panoramic Images for Investigations in Forest Applications and Tree Species Recognition. In *Proceedings of the ISPRS Working Group V/1 "Panoramic Photogrammetry Workshop"*, Dresden, Germany, 2004.
43. Liang, X.; Litkey, P.; Hyyppä J.; Kaartinen, H.; Vastaranta, M.; Holopainen, M. Automatic Stem Location Mapping Using TLS for Plot-Wise Forest Inventory. In *Proceedings of the SilviLaser 2009*, College station, TX, USA, 2009.
44. Lalonde, J.-F.; Vandapel, N.; Huber, D.F.; Hebert, M. Natural Terrain Classification Using Three-Dimensional Ladar Data for Ground Robot Mobility. *J. Field Robot.* **2006**, *23*, 839-861.
45. Munoz, D.; Bagnell, J.A.; Vandapel, N.; Hebert, M. Contextual Classification with Functional Max-Margin Markov Networks. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami Beach, FL, USA, 2009.
46. Kukko, A. *Road Environment Mapper—3D Data Capturing with Mobile Mapping*. Licentiate's Thesis. Helsinki University of Technology, Espoo, Finland, 2009.
47. Kukko, A.; Andrei, C.-O.; Salminen, V.-M.; Kaartinen, H.; Chen, Y.; Rönnholm, P.; Hyyppä H.; Hyyppä, J.; Chen, R.; Haggrén, H.; Kosonen, I.; Čapek, K. Road Environment Mapping System of the Finnish Geodetic Institute—FGI Roamer. *Int. Arch. Photogramm. Remote Sens.* **2007**, *XXXVI-3/W52*, 241-247.
48. Hyötyniemi, H. *Multivariate Regression—Techniques and Tools*. Report 125. Helsinki University of Technology, Control Engineering Laboratory: Helsinki, Finland, 2001.
49. Sithole, G.; Vosselman, G. Automatic Structure Detection in a Point-Cloud of an Urban Landscape. In *Proceedings of the 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, Berlin, Germany, 2003.
50. Jiang, X.; Bunke, H. Fast Segmentation of Range Images into Planar Regions by Scan Line Grouping. *Mach. Vis. Appl.* **1994**, *7*, 115-122.
51. Jiang, X.; Bunke, H.; Meier, U. High-Level Feature Based Range Image Segmentation. *Image Vis. Comput.* **2000**, *18*, 817-822.
52. Hoover, A.; Jean-Baptiste, G.; Jiang, X.; Flynn, P.; Bunke, H.; Goldgof, D.; Bowyer, K.; Eggert, D.; Fitzgibbon, A.; Fisher, R. An Experimental Comparison of Range Image Segmentation Algorithms. *IEEE Trans. Pattern. Anal. Mach. Intell.* **1996**, *18*, 673-689.
53. Helldén, U. *A Test of Landsat-2 Imagery and Digital Data for Thematic Mapping, Illustrated by an Environmental Study in Northern Kenya*. Rapportur och Notiser 47. Lunds Universitets Naturgeografiska Institution: Lund, Sweden, 1980.
54. de la Escalera, A.; Armingol, J.M.; Mata, M. Traffic Sign Recognition and Analysis for Intelligent Vehicles. *Image. Vis. Comput.* **2003**, *21*, 247-258.