Automatic lane marking extraction from point cloud into polygon map layer

David Prochazka, Jana Prochazkova and Jaromir Landa

Department of Informatics, Mendel University in Brno, Brno, Czech Republic; Institute of Mathematics, Faculty of Mechanical Engineering, Brno University of Technology, Brno, Czech Republic

ABSTRACT
Optimization of road networks is a common concern worldwide, primarily for safety purposes. Because the extent of these networks is substantial, automation of their inventory is highly desirable. This paper concentrates on the road inventory process that is necessary for regular maintenance. The key part of our road marking detection and reconstruction is based on spanning tree usage. The spanning trees are obtained from alpha shapes of the detected road markings. The spanning trees application enables the reliable identification of the road markings and precise reconstruction of their contours even with noisy data. Our method processes the point cloud data obtained from LiDAR measurements, and provides a common vector layer with road lane polygons. Such a vector layer is stored in a common file format supported by the majority of geographical information systems, thus producing an output that can be conveniently used for decision-making based on the road inventory process.

Introduction
Remote sensing and mobile mapping provide spatial data items that are the basis for the decision-making process in various applications. Particularly, detection of objects on the roads and in their vicinity, is one of the important areas in the last years. We can see a general effort to optimize the road network, especially the effort to make it safer. The European Union initiatives can be taken as an example. The communication "Towards a European Road Safety Area" (COM, 2010) moved the target date for halving the number of fatal road accidents forward to 2020 and set the year 2050 as the target date for moving close to having zero fatalities.

Concerning road safety, there are two key elements: cars and the road itself. The improvement of the cars is focused especially on various real-time safety systems: lane guidance, collision warning, etc. The Intelligent Transport Systems theme and the related eSafety initiative are focused on this issue. It is possible to see a substantially rising amount and quality of these systems in contemporary cars.

On the other hand, there is an improvement in the road infrastructure. Intelligent Roads, Sustainable Surface Transport and similar EU projects are focused on the road capacity increase, improvement of road maintenance and safety. For the implementation of the proposed improvements, we must have state-of-art information about the roads. It is crucial to be familiar with the state of traffic signs, road surface markings, zebra crossings (Arias, Riveiro, Soilán, Díaz-Vilarino, & Martínez-Sánchez, 2015) and even objects along the roads, for example poles or trees (Elhinney, Kumar, Cahalane, & McCarthy, 2010; Gonzalez-Jorge, Puente, Riveiro, Martínez-Sanchez, & Arias, 2013; Pu, Rutzinger, Vosselman, & Elberink, 2011).

This process of road inventory comprises a substantial amount of manual work. The human operators measure objects in the field, process large amounts of, for example aerial images or LiDAR point clouds and prepare vector representation of required objects. This process is time consuming and therefore slow and expensive. There can be found many projects aimed at automatic identification of objects related to roads (e.g. above-mentioned traffic signs or road markings). Their review is provided below. A substantial number of them are optimized for real-time car safety systems.

Our project is focused on the inventory process issue. The goal is to provide an exact vector representation stored in a format that can be processed in common geographical information systems (e.g. ESRI ShapeFile map layer). The proposed method is not suitable for real-time applications such as car safety systems mentioned. It can produce map layers that can be effortlessly used for different spatial analyses.

We proposed a three phased application that automatically process point cloud data. Firstly, we make the ground point detection using our published algorithm in Landa, Prochazka and Stastny (2013). The algorithm is based on a dynamic bounding box principle. The second step is the lane marking identification that consists of the
successive application of the alpha shape method and spanning tree. In the last step, we present a novel geometric method that is able to reconstruct arbitrary shape of the lane marking in vector form. It can process the input $k$-ary spanning tree, where $k = 1, 2, 3, 4$.

In this paper we cover the issue of automatic road line detection, identification of its kind, reconstruction of its correct shape and export to the ESRI ShapeFile map layer.

**Literature overview**

Our method, as well as methods used in similar projects, is based on the processing of point clouds provided by aerial or ground vehicles equipped with LiDAR.

Airborne laser scanning is frequently used for large areas. The airborne laser scanning data is used for digital elevation models (Sithole & Vosselman, 2005; Susaki, 2012) and applications such as monitoring of atmospheric aerosols in ecology (Badarinath, Kharol, & Sharma, 2009), structural mapping in geology (Greiby, Cunningham, Naden, & Tansey, 2012). On the other hand, the mobile laser scanning (MLS) is used for urban areas mapping (Graham, 2010) because it usually provides better results due to a higher point cloud density.

Within the urban areas, a very common goal is detection of different street objects (buildings, road surface markings, street lights etc.). These detected objects are then represented as 3D models (Lafarge & Mallet, 2011) or map layers that are used for spatial analyses (Landa & Ondrousek, 2016). The authors (Yang, Dong, et al. 2017) presents a robust method for road facilities recognition based on multiple aggregation levels computation and designing a series of contextual features to improve the recognition performance. Our article is focused particularly on detection of the road surface markings; therefore, we present primarily papers focused on this problem.

The process of detection of common street objects from LiDAR data usually starts with the classification of the terrain points (Chen et al., 2009). The classification of terrain points is used either for the isolation of objects on the ground, or for the elimination of points that are not necessary for the computations. If the target of the detection is an object with high reflectance property, then the points with high reflectance are isolated (Yang, Fang, Li, & Li, 2012).

Subsequent operations are the road marking position detection and identification of its kind, for example full or broken line (Chen et al., 2009; Yang et al., 2012). The last part of the process can be the reconstruction of road marking shape. However, this part is usually not used in methods similar to the one presented in this paper. The process commonly ends with the identification of the marking type.

**Terrain point classification**

Terrain point classification methods can be divided into two groups. The first possibility is to detect directly the terrain points (e.g. the road points), the other is to detect the adjacent road curbs. This second approach has an obvious limitation. It is not suitable for roads that are not surrounded by detectable borders.

A naive direct detection approach takes a given percentage of lowest points in the point cloud and classifies them as the ground points (Babahajiani et al., 2014). The disadvantage of this method is again clearly visible – only road segments with negligible elevation difference can be processed. Nonetheless, more complex methods are used in the practically oriented projects. In the article by Belton and Bae (2010), road points are defined as the lowest horizontal points on a smooth surface. The approach described by Yang and Dong (2013) uses a shape-based segmentation method. The segments of the point cloud are then classified using Support Vector Machines. The algorithm can detect lines or planar and spherical patches; therefore, it can be easily used for ground point detection. Another innovative method presents the application of Hough transformation on Millimetre Wave Radar data to obtain road edges (K.Y. Guo, Hoare, Jasteh, Sheng, & Gashinova, 2015).

As mentioned, the other possibility how to detect a road is to extract its road curbs. The road curbs create a boundary of the road because of their higher elevation above the road surface. For example, the method of (Yang, Fang, & Li, 2013) describes road cross-sections with window operators that filter out non-ground points. The window operators work with three criteria: elevation jump, point density and slope change. Guan et al. (2014) also uses the same criteria but for a preprocessing of the raw point cloud. The cloud is partitioned into a set of horizontal segments (so-called profiles) according to the vehicle trajectory.

**Road lane marking detection and identification from point clouds**

Traffic sign detection, as well as road surface marking detection, works with the high reflectance intensity (higher retroreflective property) of the special sign paint. A transformation of the point cloud into 2D images is commonly used.

In the article (Chen et al., 2009), the authors filter the point cloud on the basis of the point reflectance. Consequently, they generate a 2D binary image. The value of each pixel is one if it corresponds to a surface marking and zero otherwise. The Hough transform is then applied to detect the lines on the road surface. Finally, they identify the road lanes using a bounding box and RANSAC.\(^2\)

\(^2\)Random Sample Consensus.
Another use of 2D images can be seen in Guan et al. (2014). The authors generate 2D georeferenced intensity images using an extended inverse-distance-weighted (IDW) approach. Further, they segment these images into road marking candidates with a point density-dependent method.

Furthermore, the authors in Yang et al. (2012) use 2D images to detect road markings. They generate a georeferenced image of the point cloud. This image is filtered on the basis of point reflectance and height. The final step is labelling of the road marking regions according to their shape and arrangement. The method incorporates related semantic knowledge (e.g. shape, pattern) of the road marking. A similar example of the road marking detection using 2D images can be found in Thuy and Leon (2010).

Another possibility is to process the LiDAR intensity and range attributes as used in Kumar, Elhinney, Lexis, and McCarthy (2014). The described algorithm generates 2D intensity raster surfaces from the LiDAR data. According to the authors, the algorithm can detect 88% of road marking points. Naturally, the point cloud data can be also combined with common RGB images to detect lane markings. Examples can be found in Huang et al. (2013), Li, Chen, Li, Shaw and Nuchter (2014).

**Road lane marking shape reconstruction**

The road lane marking shape (envelope) is frequently required during the lane identification process. One of the most frequently used representations is a common bounding box. However, such representation has two major limitations: (a) It can be used solely for the straight lanes. (b) It can be used only in situations where the lane markings do not touch or cross each other (e.g. not on road intersections).

Another frequently used representation is the concave or convex hull (Moreira & Santos, 2007). The hull quality directly depends on the amount of noise and complexity of the shape. A solution proposed by Schindler, Maier and Janda (2012) uses circular arc splines. Nevertheless, this approach is not suitable for line intersections (see Figure 1).

The road lane marking is firstly represented by a bounding box in Chen et al. (2009). Further there is applied the RANSAC curve fitting algorithm published in Fischler and Bolles (1981) to localize each lane marking accurately. Finally, a point is selected every 10 cm along a fitted curve. The final coordinate of the point is an average of points within a 10 by 10 cm rectangular box surrounding the given selected point. The authors, however, do not reconstruct the representation to obtain the marking envelope shape, they are focused only on its detection.

The shape representations described above can have insufficient quality. Therefore, we focus on a method that allows to reliably detect and correctly reconstruct the lane marking shape in this article.

**Implementation**

This section describes our approach towards lane marking detection, identification and reconstruction. Primarily, we focus on the identification of full and broken road lane markings. A key issue is to find a precise polygon representation of each detected marking and store this representation into a common polygon map layer.

The detection process is the application of our method proposed in Landa et al. (2013). The following phase is the classic segmentation of the standard well-known method. We proposed a new two-step identification phase that consists of the alpha shape and spanning tree. The reconstruction also presents a novel geometric method how to compute the accurate shape of the lane markings. The entire process is briefly outlined in Figure 2.

**Ground point classification and lane marking detection**

Road marking detection methods are closely connected with the point reflectance as described in the previous section. Our method is based on such a common approach where the points with the reflectance higher than a given threshold are chosen. In this set of points with high reflectance values, the ground points are then classified. The ground point classification is performed solely on this set of points with high reflectance to minimize the computation needs. The reflectance value depends on the type of the scanning device and it is set experimentally.

For ground point detection, we use the algorithm proposed in Landa et al. (2013). The algorithm is
based on a dynamic bounding box principle. The input point cloud is divided into separate columns and in each column the lowest points are extracted.

The extracted ground points are further segmented on the basis of the Euclidean distance. The points satisfying empirically the determined limit of maximal distance between two points are considered as belonging into a single segment Figure 3.

The result of the Euclidean-based segmentation contains also a substantial amount of noise segments, for example building parts, curbs etc. The elimination of these false-positive segments is based on multi-threshold criteria. We determined these conditions:

- the minimal number of points in the segment
- the maximal number of points in the segment
- the maximal size of envelope rectangle in x or y direction,
- the minimal percentage of planar points using RANSAC algorithm (usual value is 95 %).

The above-mentioned criteria create the result visible in Figure 3.

**Lane marking identification**

The result of the ground point extraction and segmentation described in the previous section is a segmented point cloud where each segment represents a possible road marking. However, it is important to mention that in spite of the previous elimination of false positive results, it still contains some segments that do not represent road markings (e.g. pavement).

In this part, we describe the identification of a particular line type (full, broken). The identification process consists of four steps:

1. The 3D point cloud is transformed into a 2D point cloud.
2. The alpha shapes of represented objects are computed.
3. Their spanning trees are determined.
4. The road lane marking segments are identified.

The first step is the point cloud transformation from a geospatial coordinate system to a local coordinate system. This step simplifies the computations. Subsequently, the concave hull is frequently computed as a shape representation in some works. Nonetheless, such a concave hull is ambiguous (see Figure 4); therefore, we compute the alpha shape which is unambiguous. The alpha shape computation algorithm is described in Edelsbrunner, Kirkpatrick and Seidel (1983).

The alpha shape is a generalization of a convex hull. The general definition in Edelsbrunner, Kirkpatrick and Seidel (1983) says that alpha-hull of set $S$ is the intersection of all closed discs with radius $1/\alpha$ ($\alpha$ is a sufficiently small but otherwise arbitrary positive real number) that contain all the points of $S$. In our case, the Delaunay triangulation is used to construct a TIN (Triangulated
Irregular Network) and the alpha shape is then created on the basis of Criterion 1.

**Criterion 1**: The length of the triangle is at least two times smaller than the median of the length of all inner triangles.

This allows us to obtain a hull that is unambiguous in contrast to the concave hull. The result of the alpha shape computation is a vector polygon representing the rough shape of the lane marking (Figure 5). However, this shape cannot be used as a road lane representation. It represents the envelope of the road lane point cloud. The actual shape of the lane marking is different. For this reason, the spanning tree of the graph that is based on the alpha shape is constructed. This process of spanning tree construction is composed of (1) thinning of the alpha shape, (2) extraction of the spanning tree, (3) smoothing and filtration of the spanning tree segments.

The thinning produces the simplified representation of the image which is topologically identical with the original alpha shape image. The Guo-Hall thinning algorithm which was published in the article (Guo & Hall, 1989) is chosen. To perform thinning, the alpha shape is transformed to a 2D image. The thinning algorithm produces an image with a set of pixels that later creates a spanning tree. A tree is a connected acyclic simple graph and the spanning tree is a spanning subgraph that is also the tree. Let $k$-ary tree be a rooted tree in which each internal vertex has no more than $k$ children. A 1-ary tree is just a path. A 2-ary tree is also called a binary tree. The examples of different $k$-ary spanning trees are in Figure 6.

The subsequent extraction of the spanning tree from the image is provided by the well-known recursive region growing algorithm (Gonzalez & Woods, 2001). The algorithm searches in the neighbourhood of a start pixel (first white pixel) for non-processed pixels. If one non-processed pixel is detected, it is added to the result structure and this point is labelled as processed. If two or more non-processed pixels are detected, each pixel is labelled as the start pixel of a new part of the lane marking segment. In the case that the algorithm does not detect any non-processed point, the search is finished and the spanning tree of the graph is taken as a result. The smoothing and filtration of segments is performed by the Ramer-Douglas-Peucker algorithm (Douglas & Peucker, 1973; Ramer, 1972).

Finally, the points of the spanning tree are transformed from the local coordinate system back to the original geospatial coordinate system. All three

![Figure 4](image-url) - Different types of correct concave hulls on a set of points. These two examples present the ambiguity of a concave hull.

![Figure 5](image-url) - Concave hull determined by the alpha shape criterion. Left: Examples of point clouds that represent lines after segmentation. Right: alpha shape result.

![Figure 6](image-url) - Possible variants of spanning trees that can be constructed from the point cloud: $k$-ary spanning tree (from left to right: $k = 1,2,3,4$).
described steps are presented in Figure 7. The results projected on an aerial map are in Figure 8.

The result of this process is a set of spanning trees which represent objects on a road or in its vicinity. We focus primarily on the identification of full and broken lane markings. Our identification process is used to identify full and broken lanes and the segments they are composed of. We proposed a modification of the region growing algorithm (Figure 9) for this purpose.

Any spanning tree that contains at least one node with the multiplicity higher than 3 is eliminated as too complex. Such spanning trees represent neither broken nor full lane. The spanning trees are then divided into two categories according to their lengths (possible broken and full lane marking segments). This division is done to simplify the computations and to ensure that the spanning trees representing different road lane markings are not connected incorrectly. Inside each category, the first spanning tree is selected and its end vertices are connected to all end vertices of all unprocessed spanning trees (see Figure 9(a)). The best possible spanning tree (with the lowest distance and angle) is chosen to be connected to the original tree (Figure 9(b)). The result is a new spanning tree which connects both the original and the selected spanning tree (Figure 9(c)). This process continues until all possible spanning trees that create a set of lane markings are connected (Figure 9(d)). The process ends when all spanning trees are processed.

Lane marking representation

We focus on the mathematical construction of the vector road lane marking representation from the spanning tree in this section. As mentioned, the goal is to obtain a common vector map layer such as ESRI ShapeFile. Such a representation is suitable for any further analytic process in frequently used geographical information systems.

We choose the solution with regard to the input $k$-ary spanning tree, where $k = 1, 2, 3, 4$.

Let $\{H_i\}$ label the input vertices and $\{T_i\}$ be a set of the computed points, $d$ the given distance of the envelope points.

1-Ary tree

Let $H_i = [h_1, h_2]$ be the vertex with multiplicity one (see Figure 10). Then the result points $T_i, T_j$ can be simply expressed as:

$$
\begin{align*}
    x &= h_1 + \frac{d}{|u_1|} u_1 \\
    y &= h_2 + \frac{d}{|u_2|} u_2
\end{align*}
$$

where the vector $u = (u_1, u_2)$ is the normal vector of the edge $H_i, H_i+1$ and $d$ the given distance.

2-Ary tree

Let $H_i$ be a vertex with multiplicity two and $H_{i-1}, H_{i+1}$ vertices on the adjacent edges (see Figure 11). The computed points $T_i$ satisfies:

$$
    d = |T_iQ| = |T_iP|
$$

and

$$
    T_iP \perp H_{i+1}H_i, \quad T_iQ \perp H_{i-1}H_i \Rightarrow \Delta H_iT_iP \approx \Delta H_iT_iQ
$$

We can derive:

$$
    |H_iP| = |H_iQ| = \frac{d}{\tan \frac{\Theta_{H_iH_{i+1}}}{2}}
$$
Hence, the coordinates of the points $P$, $Q$ can be expressed using Equation (1). The result point $T_i$ can be written as the intersection of two lines $p$, $q$ perpendicular to the edges $H_{i-1}H_i$, $H_{i+1}H_i$ in points $P = [p_1, p_2]$, $Q = [q_1, q_2]$:

$$p : n_P^1 x + n_P^2 y - p_1 n_P^1 - p_2 n_P^2 = 0 \quad (3)$$

$$q : n_Q^1 x + n_Q^2 y - q_1 n_Q^1 - q_2 n_Q^2 = 0 \quad (4)$$

where $n_P$, $n_Q$ are normal vectors of lines $p$, $q$. Furthermore, the symmetry point $T_2^i$ can be simply derived by central inversion with the centre $H_i$.

### 3.4-Ary tree

Suppose that the vertex $H_i$ is connected with the vertices $H_{i+1}, H_{i+2}, H_{i+3}$ (and $H_{i+4}$ for multiplicity four). First, the order of edges has to be determined. Without loss of generality, assume that $H_iH_{i+1}$ is the
The angle $\angle H_i H_{i+1} H_k$ can be computed by:

$$\cos(\alpha) = \frac{(u, v)}{|u| \cdot |v|}$$  \hspace{1cm} (5)

$$\cos(\alpha_k) = \frac{(H_i H_{i+1}, H_i H_k)}{|H_i H_{i+1}| \cdot |H_i H_k|} \quad k = i + 2, i + 3, \text{ resp. } i + 4. \hspace{1cm} (6)$$

Because of the cosine function properties, edges can be ordered in dependence on the size and the sign of results in Equation (6). Three different situations can be distinguished (see Figure 12).

Thus, the problem is transformed to the evaluation of three (or four) couples of edges as in the case of a 2-ary tree (see Equations (2, 3, 4)). In the final step, the points are ordered and stored in a common file format. The result map layer presented above aerial imagery is in Figure 13.

**Map layers’ creation**

Two map layers are created after the shape envelope reconstruction. The first layer (polyline type) contains all identified road lane markings where each line provides information about its type (full or broken). This polyline file is suitable especially for analytical purposes. It provides information about the type and position of the markings; therefore, it can be used, for example for road safety analysis. The second created layer contains polygons of all reconstructed lane markings. This polygon file is useful mainly for the mentioned inventory process. For instance, it allows to calculate an approximate amount of paint required for the road marking maintenance and other related tasks. We have chosen the ESRI ShapeFile for polygon and polyline storage; however, any polygon or polyline format can be used. Both files are created using the Shape C Library.\(^3\)

**Results**

The evaluation is based on three different accuracy metrics: precision, recall and quality (Boyko & Funkhouser, 2011). The ground truth data pieces for the matrices are obtained with visual control of images taken during the process of point cloud capturing. The matrices are defined:

**Precision:**

$$p = \frac{TP}{TP + FP}$$

**Recall:**

$$r = \frac{TP}{TP + FN}$$

**Quality:**

$$q = \frac{TP}{TP + FN + FP}$$

where $TP$ is the number of true positives; $FP$ is the number of false positives and $FN$ is the number of false negative results.

---

\(^3\)http://shapelib.maptools.org.
We tested our method on two sets of point clouds. Both sets were obtained through mobile mapping. The first set represents an urban area inside the city centre of Brno. It was obtained using the Riegl VMX 250 Mobile Laser Scanning System. The length of the captured road is approximately 0.5 km and it contains approximately 20 mil. points. Weather conditions were good and there were no cars occluding the road lanes during the testing.

The second set represents a country road near Semice. The length of the captured road is approximately 4.3 km and it contains approximately 62 mil. points. The set was captured using the Riegl VMX 450 Mobile Laser Scanning System.

**Source point cloud data**

Figure 12. Different types of edge layout.

Figure 13. Left: individual lane marking polygons mapped on the orthophoto. Right: identified road lane markings mapped on the orthophoto. Broken lane markings are labelled in red and full lane markings are labelled in green.

---

4 [http://www.gb-geodezie.cz/wp-content/uploads/2016/01/DataSheet_Riegl_VMX-250.pdf](http://www.gb-geodezie.cz/wp-content/uploads/2016/01/DataSheet_Riegl_VMX-250.pdf).

5 [http://www.riegl.com/uploads/tx_pxpriegldownloads/DataSheet_VMX-450_2015-03-19.pdf](http://www.riegl.com/uploads/tx_pxpriegldownloads/DataSheet_VMX-450_2015-03-19.pdf).
The proposed algorithm was tested on these two datasets, but it is possible to process arbitrary data, only the threshold values should be set properly. Due to the different type of scanning devices and different point clouds, the appropriate threshold parameters are important. The size of the point cloud is limited only by the computational capacity of the computer. Authors (Yang, Liu, et al., 2017) presented the local feature binary kernel descriptor (BKD) that encodes the shape and intensity information of point cloud data. BKD descriptor increases the robustness and noise independence.

We used MacBook Air, Intel i5 1.7Ghz, 4GB RAM, SSD drive without the use of multi-threading optimization for the testing. Our application is written in C++ and uses several open-source libraries: PCL, OpenCV, libLAS and Shape C Library. The Point Cloud Library (PCL) is used for common point cloud processing operations, such as segmentation and visualization. The OpenCV library is used for image and 2D data processing, such as triangulation and thinning. Both PCL and OpenCV are chosen because of their wide functionality and community. The libLas library helps with LAS file loading. Finally, the Shape C Library exports the data to ESRI ShapeFile.

**Lane marking identification**

The resultant spanning trees can be categorized into three categories based on their properties, such as length and k value:

- spanning trees of broken lane marking segments,
- spanning trees of full lane marking segments,
- other spanning trees.

**First set results (urban area)**
The first set contains the total number of five broken and five full lines. Out of the total number of 48 broken line segments, 46 segments were correctly identified. Two broken line segments were missed during the identification process. All full line segments were identified. The precision of broken lines identification is equal to 98.45% and its confusion matrix is shown in Table 1.

**Second set results (country round)**
The tested area contains the total number of 12 broken and 10 full lines. The full lines are composed of 72 line segments and only one solid line segment is missed due to the paint degradation. The results are in Figure 15.

The 12 broken lines consist of 287 road line segments. The method identifies 264 of them correctly (see Figure 14). 19 spanning trees are identified incorrectly as road lane marking segments. Nevertheless, the accuracy reaches 90 per cent. The main reason for the incorrect identification is that these false segments have similar shape to the road lines and they are positioned next to each other. The confusion matrix for these results is in Table 2.

**Time consumption of the method**
The time consumption for different parts of the method is presented in the Table 3. As for the first set, the entire process took approximately 3 min. As for the second set, the entire process took 138 min.

---

**Table 1. Confusion matrix for broken lane markings from the first data set; positive predictive value (PPV), false omission rate (FOR), true positive rate (TPR), false positive rate (FPR), accuracy (ACC).**

| Predicted values | 48 | 46 | 2 |
|------------------|----|----|---|
| ACC = 98.48 %    | 84 | 0  | 84|
| PPV = 100.0%     | FNR = 4.17 % |
| FOR = 2.32%      | TNR = 100.0% |

6[http://pointclouds.org](http://pointclouds.org).
7[http://opencv.org](http://opencv.org).
8[http://www.liblas.org](http://www.liblas.org).
9[http://shapelib.maptools.org](http://shapelib.maptools.org).

---

**Figure 14.** The result of the identification process. The red and green colour represents correctly identified broken and full road lane markings. The yellow colour represents incorrectly identified broke lane marking (false positive).
The least time-consuming part of the process is the road lane marking envelope reconstruction. In case of the first set of point clouds, the most time-consuming part is the spanning tree computation. This is due to the higher complexity of segments computed during the segmentation. On the other hand, in the second set, the most time-consuming part of the process is the identification. The main reason is the high number of simple spanning trees.
Discussion

The article presents a new method for road lane marking detection, identification and its shape reconstruction from the LiDAR data. Our method is different from other approaches because it is aimed at the exact representation of road lane markings for analytical purposes. During the testing, the proposed method showed potential for fully automated road marking processing. However, the testing also revealed some limitations. The method is not suitable for use in complicated road intersections.

Another limitation is the dependence on the detection part of the method. The detection part produces a large number of false positives. The reason is the limitation of the used ground point classification. We select all ground points and not only the road surface points as can be seen in Belton and Bae (2010), Yang et al. (2013). False positive decrease is the important task to future. If we can detect only all ground points, we can reduce the false positive significantly.

The identification part of the method is based solely on the common properties of the road markings (e.g. length). We do not use any other information, such as a car path as can be seen in Thuy and Leon (2010). Thanks to it, we can identify all road lane markings and not only the lines that are parallel to the car path. The identification focuses on both full and broken lines as opposed to Yang et al. (2012). However, the identification process fails in case when several spanning trees of consecutive road lane marking segments are not created. In these cases, the identified lane is split into two parts. Future development will focus on this issue.

The shape reconstruction is usually based on the direct processing of the point cloud data. As mentioned, we can find many different approaches. The point cloud can be connected with standard RGB images. In this case, it is possible to create bounding boxes around identified markings and apply the RANSAC curve fitting to localize the correct lane marking position within the image (Chen et al., 2009). A disadvantage of this approach is that the method needs RGB images correctly paired with the point cloud data. Moreover, this method is not able to deal with noise on the shape boundaries. It is possible to apply the Douglas-Peucker line simplification algorithm (Wu & Marquez, 2003), which can be used for obtaining a non-self-intersection polygon envelope. However, complexity rejection must be used for exclusion of too complex shapes. These too complex shapes frequently originate in the noise. Computational complexity is $O(mn)$ where $n$ denotes the number of input vertices and $m$ the number of output segments. In contrast to the described approaches, our method does not need any interconnection with RGB images; further, it can be applied on all shapes including complex shapes (e.g. arrows). If the marking is found, we obtain its spanning tree. Using spanning trees has two clear advantages. First, the trees can be used easily to identify the kind of road lane marking. Secondly, they can be used for reconstruction of the road lane marking envelope, as proposed in this article. Therefore, our envelopes are not influenced by the noise.

The challenge of our algorithm is the analysis of the complex shapes, for example the crossing lines on crossroads or non-straight shapes. Also, there is big challenge with the lanes with missing colour.

Conclusion

We present a novel algorithm for autonomous reconstruction of road lane markings, specifically broken and full road lane markings, from LiDAR data. We start with the detection based on point reflectance values. Using the standard Euclidean distance segmentation, the point cloud is divided into the potential road lane markings.

Afterwards, the method continues with the identification phase where each possible road lane marking is transferred to a spanning tree representation. The spanning trees offer a much better lane marking shape representation than the commonly used bounding box, convex or concave hull. The spanning tree representation also simplifies the identification process. The road lane markings are identified using a region growing algorithm.

The last phase, lane marking shape reconstruction, comprises a geometric approach. Each spanning tree is encased in a polygon envelope in a given distance. This envelope can be stored as a common polygon map layer, in our case an ESRI ShapeFile that can be used in virtually any GIS application. Therefore, the proposed method is not focused on a single issue. It covers all necessary steps for production of a common map layer from a given raw point cloud.

Our approach is very reliable as shown in the results. The first tested area is located in the Brno city centre in the Czech Republic, particularly in the vicinity of Mendel Square. It comprises common streets and squares. The roads contain multiple road lanes for cars as well as separate lanes for trams and trolleybuses. The second tested area is the road near Semice town in the Czech Republic. Therefore, the tested areas are completely different. Although our method produces a substantial number of false positives during the detection phase, the identification phase eliminates vast majority of them. The true positive rate is in all cases above 92 %. This indicator is the most important from the road inventory point-of-view because small amount of false positive results can be easily removed by a human operator.
Our key advantage is the ability to reconstruct the precise shapes of the road lane markings. The reconstructed shapes can be used for the visualization as well as for analytic purposes (maintenance cost calculation). The results are provided in two map layers based on ESRI ShapeFile standard. The first layer contains polylines with the kind of the road lane markings, the other contains the precise reconstructed shapes of the markings.

Acknowledgements
The first data set (Brno) used in this article was provided by GEODIS BRNO, s.r.o. The second data set (Central Bohemian Region) was provided by the consortium of companies: CleverMaps, Consultest, AdMaS and Geodrom.

Disclosure statement
No potential conflict of interest was reported by the authors.

ORCID
Jana Prochazkova @ http://orcid.org/0000-0002-7963-1942

References
Arias, P., Riveiro, B., Soilán, M., Díaz-Vilarino, L., & Martínez-Sánchez, J. (2015). Simple approaches to improve the automatic inventory of zebra crossing from mls data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-3/W3, 103–108. doi:10.5194/isprs-archives-XL-3-W3-103-2015
Babahajiani, P., Fan, L., & Gabbouj, M. (2014). Object recognition in 3D point cloud of urban street scene. In C.V. Jawahar & S. Shan (Eds.), Computer vision - ACCV 2014 workshops (pp. (177–190)). ACCV 2014. Cham: Springer.
Badarinath, K., Khoral, S.K., & Sharma, A.R. (2009). Long-range transport of aerosols from agriculture crop residue burning in indogangetic plains - a study using LIDAR, ground measurements and satellite data. Journal of Atmospheric and Solar-Terrestrial Physics, 71(1), 112–120. doi:10.1016/j.jastp.2008.09.035
Belton, D., & Bae, K.H. (2010, June). Automating post-processing of terrestrial laser scanning point clouds for road feature surveys. In J.P. Mills, D.M. Barber, P.E. Miller, & I. Newton, (Eds.), ISPRS 2010 Proceedings of the ISPRS Commission V Mid-Term Symposium on Close Range Image Measurement Techniques (74–79). Newcastle upon Tyne, UK: ISPRS.
Boyko, A., & Funkhouser, T. (2011). Extracting roads from dense point clouds in large scale urban environment. ISPRS Journal of Photogrammetry and Remote Sensing, 66(Suppl. 6), S2–S12. doi:10.1016/j.isprsjrs.2011.09.009
Chen, X., Stroila, M., Ruisheng, W., Kohlmeyer, B., Narayanan, A., & Jeff, B. (2009, November). Next generation map making: Geo-referenced ground-level LIDAR point cloud for automatic retro-reflective road feature extraction. Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 488–491. Seattle, USA: ACM.

COM. (2010). Towards a European road safety area: Policy orientations on road safety 2011–2020 (Report No. 2010/ 0903), European Commission.
Douglas, D., & Peucker, T. (1973). Algorithm for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica, 10(2), 112–122. doi:10.3138/FM57-6770-U75U-7727
Edelsbrunner, H., Kirkpatrick, D., & Seidel, R. (1983). On the shape of a set of points in the plane. IEEE Transactions on Information Theory, 29(4), 551–559. doi:10.1109/TIT.1983.1056714
Elinnney, C.M., Kumar, P., Cahalane, C., & McCarthy, T. (2010). Initial results from European road safety inspection (euri5) mobile mapping project. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 38(S), 440–445.
Fischler, M.A., & Bolles, R.C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communicable ACM, 24(6), 381–395. doi:10.1145/358669.358692
Gonzalez, R.C., & Woods, R.E. (2001). Digital image processing (2nd ed.). Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
Gonzalez-Jorge, H., Puente, I., Riveiro, B., Martinez-Sanchez, J., & Arias, P. (2013). Automatic segmentation of road overpasses and detection of mortar efflorescence using mobile LIDAR data. Optics & Laser Technology, 54, 353–361.
Graham, L. (2010). Mobile mapping systems overview. Photogrammetric Engineering and Remote Sensing, 76(3), 222–228.
Grebbey, S., Cunningham, D., Naden, J., & Tansey, K. (2012). Application of airborne LIDAR data and airborne multispectral imagery to structural mapping of the upper section of the Troodos ophiolite, Cyprus. International Journal of Earth Sciences, 101(6), 1645–1660. doi:10.1007/s00531-011-0742-3
Guan, H., Li, J., Yu, Y., Wang, C., Chapman, M., & Bisheng, Y. (2014). Using mobile laser scanning data for automated extraction of road markings. ISPRS Journal of Photogrammetry and Remote Sensing, 87, 93–107. doi:10.1016/j.isprsjrs.2013.11.005
Guo, K.-Y., Hoare, E., Jasteh, D., Sheng, X.-Q., & Gashinova, M. (2015). Road edge recognition using the stripe Hough transform from millimetre-wave radar images. IEEE Transactions on Intelligent Transportation Systems, 16(2), 825–833.
Guo, Z., & Hall, R.W. (1989). Parallel thinning with two-subiteration algorithms. Communications of the ACM, 32(3), 359–373. doi:10.1145/62065.62074
Huang, H., Wu, S., Cohen-Or, D., Gong, M., Zhang, H., Li, G., & Chen, B. (2013). L1-medial skeleton of point cloud. ACM Trans. Graph., 32(4), 65:1–65: 8. doi:10.1145/2461912.2461913
Kumar, P., Elinnney, C.M., Lekis, P., & McCarthy, T. (2014). Automated road markings extraction from mobile laser scanning data. International Journal of Applied Earth Observation and Geoinformation, 32, 125–137. doi:10.1016/j.jag.2014.03.023
Lafarge, F., & Mallet, C. (2011). Building large urban environments from unstructured point data. In 2011 IEEE International Conference on Computer Vision (1068–1075). Barcelona, Spain: IEEE.
Landa, J., & Ondrousek, V. (2016). Detection of pole-like objects from LIDAR data. In 19th International Conference Enterprise and Competitive Environment 2016 (226–235). Brno, Czech Republic: Mendel University.
Landa, J., Prochazka, D., & Stastny, J. (2013). Point cloud processing for smart systems. Acta Univ. Agric. Silvic.
Mendel. Brun., 61(7), 2415–2421. doi:10.11118/actaun201361072415

Li, Q., Chen, L., Li, M., Shaw, S.-L., & Nuchter, A. (2014). A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios. IEEE Transactions on Vehicular Technology, 63(2), 540–555.

Moreira, A., & Santos, M. (2007, March). Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. In GRAPP 2007: Proceedings of the International Conference on Computer Graphics Theory and Applications (61–68). Barcelona, Spain: INSTICC.

Pu, S., Rutzinger, M., Vosselman, G., & Elberink, S.O. (2011). Recognizing basic structures from mobile laser scanning data for road inventory studies. Journal of Photogrammetry and Remote Sensing, 66(6), S28–S39. doi:10.1016/j.isprsjprs.2011.08.006

Ramer, U. (1972). An iterative procedure for the polygonal approximation of plane curves. Computer Graphics and Image Processing, 1(3), 244–256. doi:10.1016/S0146-664X(72)80017-0

Schindler, A., Maier, G., & Janda, F. (2012, June). Generation of high precision digital maps using circular arc splines. In 2012 IEEE Intelligent Vehicles Symposium (IV) (246–251) Madrid, Spain: IEEE.

Sithole, G., & Vosselman, G. (2005). Filtering of airborne laser scanner data based on segmented point clouds. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36(3), 66–71.

Susaki, J. (2012). Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) generation. Remote Sensing, 4(6), 1804–1819. doi:10.3390/rs4061804

Thuy, M., & Leon, F.P. (2010). Lane detection and tracking based on LiDAR data. Metrology and Measurement Systems, 17(3), 311–321. doi:10.2478/v10178-010-0027-3

Wu, S.T., & Marquez, M.R.G. (2003, October). A non-self-intersection Douglas-Peucker algorithm. In XVI Brazilian Symposium on Computer Graphics and Image Processing (60–63) SIBGRAPI 2003. Sao Carlos, Brazil: IEEE.

Yang, B., & Dong, Z. (2013). A shape-based segmentation method for mobile laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 81, 19–30.

Yang, B., Dong, Z., Liu, Y., Liang, F., & Wang, Y. (2017). Computing multiple aggregation levels and contextual features for road facilities recognition using mobile laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing, 126, 180–194. doi:10.1016/j.isprsjprs.2017.02.014

Yang, B., Fang, L., & Li, J. (2013). Semi-automated extraction and delineation of 3D roads of street scene from mobile laser scanning point clouds. Journal of Photogrammetry and Remote Sensing, 79(4), 80–93. doi:10.1016/j.isprsjprs.2013.01.016

Yang, B., Fang, L., Li, Q., & Li, J. (2012). Automated extraction of road markings from mobile LiDAR point clouds. Photogrammetric Engineering and Remote Sensing, 78(4), 331–338. doi:10.14358/PERS.78.4.331

Yang, B., Liu, Y., Dong, Z., Liang, F., Li, B., & Peng, X. (2017). 3D local feature BKD to extract road information from mobile laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 130, 329–343. doi:10.1016/j.isprsjprs.2017.06.007