Automation in Construction 51 (2015) 103–112

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

An approach to detect and delineate street curbs from MLS 3D point cloud data

Borja Rodríguez-Cuenca [a],*, Silverio García-Cortés [b], Celestino Ordóñez [b], Maria C. Alonso [a]

[a] Department of Physics and Mathematics, University of Alcalá, Campus Universitario Ctra. Madrid-Barcelona, km. 33,600, 28871 Alcalá de Henares, Madrid, Spain
[b] Department of Mining Explotation, University of Oviedo, Escuela Politécnica de Mieres, C/Gonzalo Gutiérrez Quirós, 33600 Mieres, Asturias, Spain

ARTICLE INFO

Article history:
Received 4 June 2014
Received in revised form 9 October 2014
Accepted 11 December 2014
Available online 15 January 2015

Keywords:
Mobile laser scanner
Curb extraction
3D point cloud
Feature extraction
Urban modeling
Segmentation

ABSTRACT

This paper is focused on the detection and delineation of curbs and street boundaries using a 3D point cloud registered by a mobile laser scanner (MLS) system. Non-manual roadside detection is an important issue in road maintenance, 3D urban modeling, and autonomous navigation fields. The proposed method is able to detect upper and lower curb edges and delineate even occluded or hidden roadsides in most environments. It is based on a rasterization and segmentation approach for curb detection while preserving the reference to the original point cloud avoiding loss of information.

The method was tested using two point clouds from various street scenes measured by an Optech Lynx Mobile Mapper System, providing accuracies in curb extraction over 95% in both datasets.

1. Introduction

The automatic detection of the various constructed elements in roads and streets has become the subject of research in recent years because of its practical interest. Accurate automatic detection saves a great deal of time and money during the creation and updating of cartographic databases [1]. Moreover, recently, there have been important advances in the development of autonomous driving systems. These systems require an accurate detection of road boundaries between which the autonomous vehicle [2,3] or pedestrian robot [4] must drive. Precise road boundary detection will increase autonomous driving systems safety and prevent accidents.

Several methods have been developed to detect roads based on imagery data. A thorough review can be found in [5,6]. Road extraction and classification from images have been under research for many years. The different approaches vary depending on the road model and road representation selection which, in turn, are directly related with the sensor resolution [7]. In low-resolution images, roads appear as thin lines. In these cases, detection is reduced to center road extraction. In high-resolution data, roads appear as two-dimensional areas rather than one-dimensional line. In these cases, the boundaries of roads will be detected instead the centerline [8]. In [1,6,9,10], some methodologies to extract roads from aerial and satellite images are presented. Many other approaches focus on road detection methods from Radio Detection And Ranging (RADAR) and Light Detection And Ranging (LIDAR) data [8,11]. Detection in urban environments could be difficult due to the occlusions produced by high elements such as trees or buildings. Those elements produce errors in detecting land covers because they appear to overlap with the real land cover in the aerial images [12] and cast shadows on the ground in laser pulse datasets.

In the last decade, several laser scanner sensors have been developed. These sensors provide unstructured data in the form of point clouds with very high densities. This data open the possibilities for an automatic feature detection of cartographic entities in urban environments. Laser scanner sensors could be placed on aerial (aerial laser scanner (ALS) or aerial LIDAR) or terrestrial platforms (terrestrial laser scanner (TLS) or terrestrial LIDAR). Terrestrial LIDAR can be categorized into two types: static and dynamic. Static terrestrial LIDAR data collection is carried out from base stations: A sensor is fixed in the base station, from which the point cloud is sensed. Dynamic terrestrial LIDAR or mobile laser scanner (MLS) sensors are installed in a mobile platform. MLS sensors have a navigation system based on global navigation satellite systems (GNSS) and inertial measurement units (IMU). These devices determine the position of the mobile platform and the direction and orientation of the sensor at every moment [13]. The GNSS and IMU data are combined with measures carried out by the laser scanner sensor in order to obtain the final product: a georeferenced point cloud [14].

Laser scanner sensors provide additional and complementary information to that provided by aerial images. TLS and MLS provide point

* Corresponding author. Tel.: +34 918 85 67 48.
E-mail address: borja.rodriguezc@edu.uah.es (B. Rodríguez-Cuenca).

http://dx.doi.org/10.1016/j.autcon.2014.12.009
0926-5805/© 2014 Elsevier B.V. All rights reserved.

clouds with a higher density than those detected with LIDAR flights (ALS), and they are useful in detecting cartographic entities in urban and rural areas. These instruments are very attractive in terms of documentation and inspection work due to the speed of data collection, their accuracy, and the fact that they do not require direct contact with the objects of interest for data collection [15].

The goal of this work is to present a new method to detect street curbs from 3D point clouds registered using a MLS sensor. These street elements are interesting for the development of autonomous driving applications and urban elements inventory purposes like pole-like elements, traffic signals, etc. This method provides a solution to this problem based on the projection of the measured point cloud on the XY plane. Over that plane, a segmentation algorithm and linear elements search is carried out in order to determine the location of street boundaries. The proposed method is valid in both straight and curved road sections. Furthermore, a solution to the problem of detecting the upper and lower edges of the curb and estimating occluded boundaries in the measured point cloud is provided. The paper is organized as follows: Section 2 summarizes the previous studies related to ours; in Section 3 the proposed method to detect street boundaries is detailed; and Section 4 shows the results obtained in two study cases, both performed in the north of Spain. Finally, our conclusions are given in Section 5.

2. Related work

Many applications for terrestrial laser scanners have been reported since the appearance of these systems. The 3D modeling of buildings, caves and indoor areas [16,17], maintaining control over soil erosion and rock fall hazards [18], the geometry verification of tunnels [19,20], the modeling and reconstruction of 3D trees [21], or the roughness soil description [22,23] are some of the applications for which TLS's have been used. Additionally, several applications for point clouds detected via MLS sensors exist in the current literature. They have been used in applications such vertical wall extraction, façade modeling, building footprint detection [24–26] and the extraction of pole-like objects, such traffic signs, lamp posts, or tree trunks [27,28].

The work presented in this paper is devoted to curb and street boundary extraction from point clouds detected by MLS sensors. In the current literature, there are some studies related to this issue. Some of them use as input data point clouds obtained from stereo vision. Recently, some authors have focused on the detection of road markings, lines, and road sides in straight and curved areas based on data obtained by stereo cameras [29,30]. Siegemund et al. [31] present a real-time method to determine and reconstruct curbs from 3D point clouds created from stereo vision or measured by a MLS. They determine the values of certain parameters about surfaces and curbs that are included in a model. Detection is carried out in an iterative procedure that involves two steps. First, the points of the 3D point cloud are assigned to curbs or adjacent surfaces; and then, surfaces and curbs are reconstructed by fitting a cubic/three-order polynomial. In [32], a robust algorithm that performs real-time lane detection and tracking based on Random Sample Consensus (RANSAC) is proposed. This algorithm works with images obtained by a camera installed on a car.

There are also some methods in the current literature to detect curbs and roadsides based on point clouds measured with TLS and MLS sensors. In [33], a method to detect curbs using 3D scanner sensor data is presented. The detection process starts with the voxelization of the point cloud and the separation of those points that represent the ground. Later, candidate points for curbs are selected based on three spatial variables: height difference, gradient value, and normal orientation. Using a short-term memory technique, every point located in a voxel whose vertical projection is in the road is considered a false positive and is deleted. Finally, the curb is detected by adjusting a parabolic model to the candidate points and performing a RANSAC algorithm to remove false positives. The performance of the method depends on the correct selection of the thresholds for each of the three variables used. This method provides a detection rate of about 98% in two studied datasets.

Weiss and Dietmayer [34] automatically determined the position of lane markings, sidewalks, reflection posts, and guardrails by a vertically and horizontally automotive laser scanner data. Curb detection applies a third-order Gaussian filter to sharpen the vertical distance profile, which defines the shape of the curb. This profile is divided into sections with a certain width, forming an accumulative histogram. Candidate curbs are found through a histogram-based algorithm to search those slots of the histogram that are candidates to represent curbs and guardrails. Because not every candidate is a valid curb, the locations of real curbs are determined by analyzing the heights, slopes, and interruptions of every polygon.

Belton and Bae [15] proposed a method to automatize the identification of curbs and signals using a few steps. The rasterization of the 3D point cloud into a 2D grid structure allows each cell to be examined separately. The method first extracts the road; then cells which are adjacent to the road are likely to contain curbs. Points in these cells are used to determine the vertical plane of the curb, from which a 2D transversal section is calculated. The top and the bottom of the curb are determined as those points which are furthermost above and below the line defined by the two furthermost points in the 2D section. This procedure has several limitations. The proposed method would not provide good results detecting concave and non-horizontal roads; furthermore, the method could provide poor results for shorter or curved curbs due to confusing edges with other points of the studied profile.

Yang et al. [35] carried out an edge detection by dividing the measured point cloud into two-dimensional sections using the GPS time at which every point was registered. They applied a moving window to these 2D sections to detect the roads and road boundaries. Curbs were detected by analyzing the elevation and shape change in the moving windows studied. They also presented a method to detect curbs in occluded parts of the cloud, but some problems in areas with irregular shapes were detected. The value of the parameters and the length of the moving window are critical to the performance of the proposed method.

A recent work by Hervieu and Soheilian [36] describes a method to extract curbs and ramps, as well as reconstruct lost data in areas hidden by obstacles in the street. A system for the reconstruction of road and sidewalk surfaces is also proposed. They adjust a plane to a group of points from the cloud and compute the angular distance between the normal vector and the z vector. After that, a prediction/estimation model is applied to detect road edges. The procedure requires the user to manually select the curb direction, which is not always easy. This method could fail in curved or occluded sections. To solve this problem, they propose a semi-automatic solution in which the user must choose some points of the non-detected curb to reconstruct these sections.

In [37] Kumar et al. developed a method to detect road boundaries in both urban and rural roads, where the non-road surface is comprised of grass and soil and the edges are not as easily defined by slope changes alone. A 2D rasterization of the slope, reflectance, and pulse width of the detected point cloud is carried out. Gradient vector flow and a balloon model are combined to create a parametric active contour model, which allows the road boundaries to be determined. Roadside detection is carried out using a snake curve, which is initialized based on the navigation track of a mobile van along the road section. The snake curve moves using an iterative process until it converges on the roadsides, where the minimum energy state is located. This method has been tested in straight sections and provided good results, but its performance in curved sections is unknown. The procedure is computationally complex, which could make the detection process too slow.

There exist some solutions for curb detection in commercial software packages [38] but unfortunately the technical details could not be found in the literature. These solutions are not automatic at all; the user must provide some initial elements to the software, especially in curved sections.

3. Method

The proposed method of detecting and delineating curbs consists of six steps. The flow chart of the procedure in Fig. 1 briefly describes every step of the method. The inputs of the process are 3D point clouds measured by the MLS sensor and vehicle trajectory points obtained from the GPS/INS systems, describing the trajectory of the vehicle. The outputs of the procedure are the points of the curbs and lines representing the curbs in occluded areas.

Every step of the proposed method is described in detail below:

3.1. Orientation and trimming

The point clouds obtained by mobile laser scanner (MLS) are properly geo-referenced in a global reference system by means of a navigation system and an IMU, which provides coordinates within a global frame to every point. In fact, the point clouds used in both study cases of this work are geo-referenced using UTM projection in Zone 29 and WGS84 ellipsoidal heights. In order to ease and speed up every operation, the original coordinate system is transformed by means of a translation and two rotations into a local Cartesian coordinate system whose origin is located at the beginning of the MLS trajectory, being the x-axis coincident with the average direction of the vehicle and the zero height plane at the height of the GNSS phase center system. It is obvious that points belonging to curbs will be placed below the new XY plane after the previous geometric transformation. Now a trimming process is carried out to reduce the point cloud size: only points with local negative z-coordinate values are kept, being removed those points located over the GPS antenna height.

3.2. Rasterization

The next step of the procedure consists of the rasterization of the point cloud to reduce its dimension and make it more manageable, moving from a 3D cloud to a 2D raster image. Moreover, in a raster image, it is possible to apply image analysis techniques to detect those pixels that contain points that belong to a roadside.

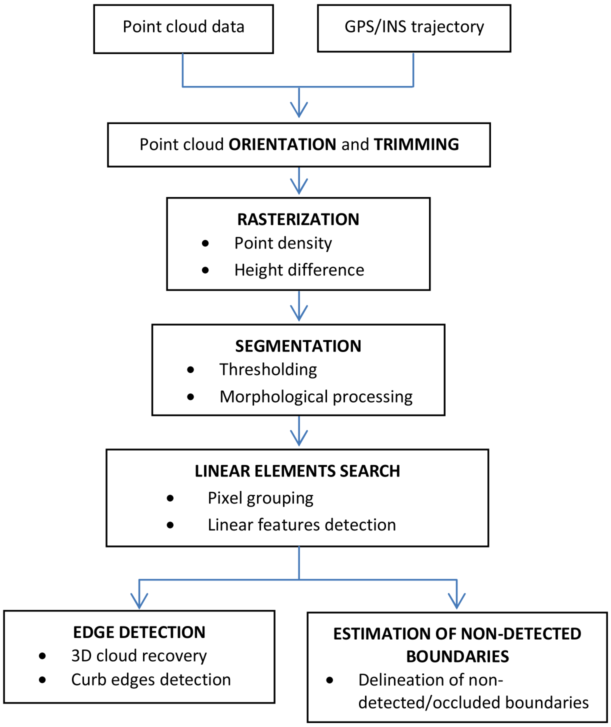


Fig. 1. Flow chart of the proposed method to detect and delineate street boundaries.

The efficiency of the detection process will depend on the pixel size of the created image. Cell size also depends on the point density of the point cloud. The resolution of the rasterized image could grow proportionately with the point cloud's density, but these also could require more computation resources. In any case, the grid spacing must be large enough to contain a significant number of points, but small enough to allow only a small number of salient features in each cell [15]. To decide the proper size of the cell, distances between contiguous pulses on the pavement—just below the central rear point of the vehicle—were measured. These points are located at a constant distance for any dataset and have been used to compare nominal densities of different cloud points. In this study the used cell size was 5 times the distance between those contiguous pulses. We have found that this value allows a proper detection of curbs with different scan densities and curb widths. For each cell in the rasterization grid, two digital values (DV) are calculated and stored: (i) the difference between the highest and the lowest point of all points contained in the studied cell (resulting an image similar to a normalized digital surface model (nDSM) [39]) and (ii) the number of points contained in every cell. After the rasterization step two images are available: one with DV representing the height difference (referred to as nDSM from now on) and another in which each pixel's DV is the number of points contained within it (referred to as image density from now on).

3.3. Segmentation

3.3.1. Thresholding

In this stage a binary image will be created by thresholding the two former images. Common curbs show heights of few tens centimeters but they can vary depending on the country and type of street. Candidate curb pixels are chosen from the nDSM image by thresholding (Eq. (1)). The maximum and minimum thresholds will be set as the maximum curb height expected and the minimum value which allow avoiding extraction of points from the road. In Table 1 can be found the threshold values used for the test datasets.

$$\text{Hmin} < \text{nDSM}[i,\ j] < \text{Hmax} \tag{1}$$

Surfaces that are orthogonal to the laser pulses show a higher density than those that are nearly parallel to the laser pulses [28,35]. Thus, pixels that represent the vertical face of the curb must have a higher DV in the density image than those that represent a horizontal surface. A second condition was imposed on the density image: Pixels that represent a curb must have a DV in the density image that is higher than an established threshold (Dmin) (Eq. (2)).

$$\text{Density image}[i,\ j] > \text{Dmin}\,(\text{points/pixel}) \tag{2}$$

With these two restrictions, a new binary image is created, where candidate pixels are flagged with 1. The rest of the detection process will affect only to these pixels. There will be more pixels, in addition to those that represent curbs, which fulfill the conditions imposed, as those that represent steps of stairways, low vegetation or cars. The

Table 1
Algorithm settings used in the test sites 1 and 2.

Pixel size	5 × 5 cm
Δh (Hmin and Hmax)	0.10 < nDSM < 0.20
Point density (Dmin)	5 points/pixel
Line width (p and q)	3 pixels < Line width < 6 pixels
Line percent (μ)	80%
Region aspect ratio (columns/rows) (m and n)	≤1/3 and ≥3

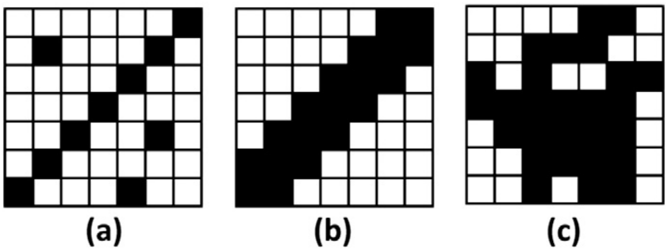


Fig. 2. (a) Synthetic image representing a linear structure and three isolated pixels, (b) Result of the morphological opening operation and (c) a nonlinear region.

following steps in the process will try to remove from the binary image those 1-value pixels which do not belong to curbs.

3.3.2. Morphological processing

A morphological opening is applied to the binary image obtained in the former step. This morphological operation is conceptually composed of two phases erosion and dilation [40]. The erosion operation will first remove those isolated pixels that do not represent a curb but satisfied the conditions established in the thresholding step. During the dilation operation, every pixel that is 4-connected with the candidate pixels of the original binary image is added to the curb candidate set of pixels being labeled with 1.

An example of the performance of the opening operation can be seen in Fig. 2. Dilation is applied to an image in which a linear structure and three isolated pixels are presented (Fig. 2(a)). The linear element represents a curb, and the isolated pixels correspond to those satisfying the rules of the thresholding step but not representing a roadside. By applying the morphological operator, this salt and pepper effect is removed and linear structure grows (Fig. 2(b)). The morphological processing removes the isolated pixels wrongly considered as candidates but not those groups of pixels that satisfy the imposed thresholds and represent other features different from curbs. These errors will be corrected in the next step through a study of the shape of the regions in the binary image.

3.4. Linear elements search

3.4.1. Pixel grouping

Those curbs that define the roadside have a linear shape different than other street elements. It is assumed that, in the binary image, pixels that represent curbs are grouped forming linear structures. Pixels that do not represent curbs are not grouped or are grouped without a particular distribution. Thus, every group of candidate pixels without a linear shape will be deleted. The linear element search starts with a grouping algorithm. Pixels with a value of 1 from the binary image that have an edge in common are grouped in a unique region. Thus, the former binary image becomes a segmented image. Every region created in this step has its own identifier.

The grouping procedure consists of an image segmentation based on a region-growing algorithm over the binary image [41].

3.4.2. Linear feature detection

In order to remove the noisy pixels, linear feature detection is carried out in every region that was created previously. A common feature of every target linear region is that its width is similar to the breadth of the real-curb line it represents. Counting the number of 1-value pixels in each column and in each row that exist in the window defined by the region, we find that most of them must have a number of pixels equal to the width of the target line. Thus, a diagonal 3 pixel width real-curb line is represented in Fig. 2(b) and (c), showing a nonlinear region, which corresponds to a wrong detected region that must be removed. Regions that represent curbs will be detected by imposing the condition that at least a certain percentage of columns or rows must have as many pixels as the width of the target line.

For each region, the number of rows and columns that have a width between p and q pixels was determined via Eqs. (3) and (4), where r and c represent the total number of rows and columns, respectively, in the studied region. Every region that has a percentage of rows or columns (rowspercent and colspercent in Eqs. (5) and (6)) with a number of pixels between the width line thresholds that are higher than a selected percentage (μ) will be considered to be a linear feature. Those regions that do not meet this requirement are discarded and not

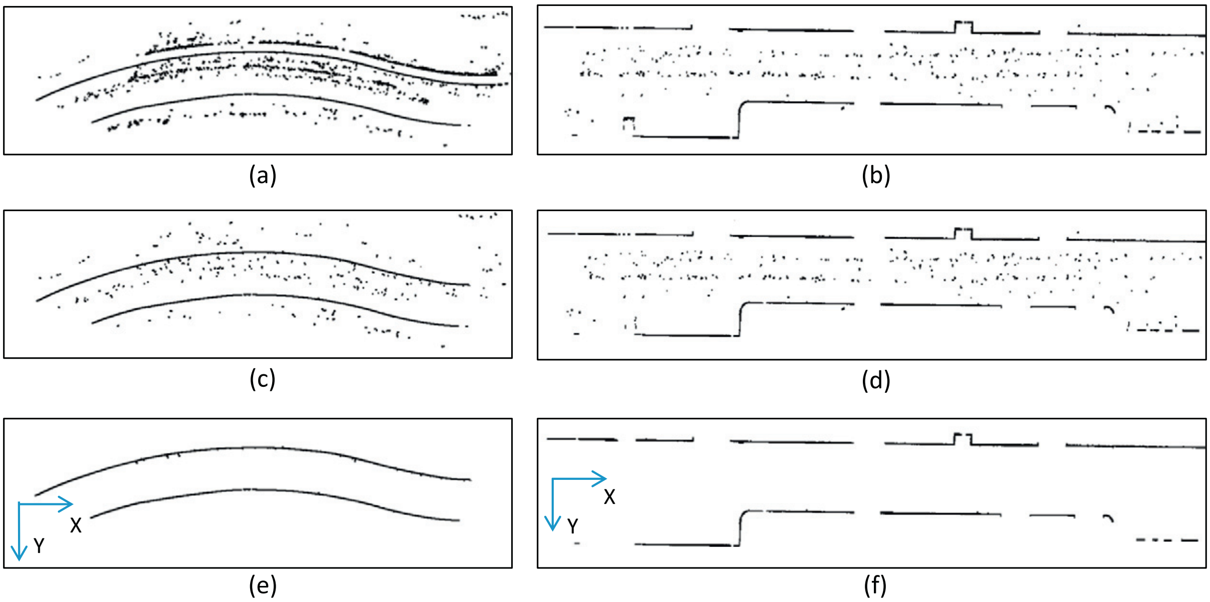


Fig. 3. Steps in the linear element detection procedure for Test Sites 1 and 2. (a) and (b) are the outputs of the threshold process, (c) and (d) are the images obtained after first step of the linear elements search, and (e) and (f) are the binary images based on curb detection.

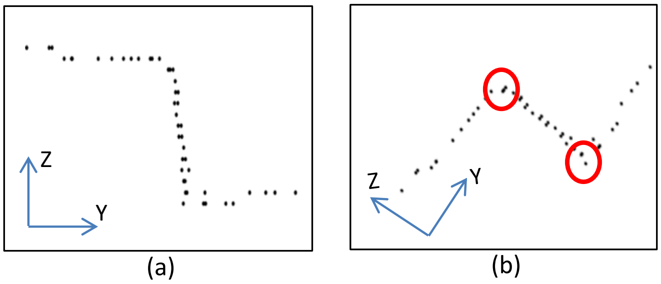


Fig. 4. (a) Profile corresponding to a curb before the rotation around the x-axis; the horizontal and vertical surfaces of the sidewalks, curbs, and roads could be easily recognized; (b) the same curb after the 45° rotation around the x-axis has been carried out; curb surfaces are now inclined +/−45°. The edges correspond with points where the slope sign changes (those surrounded by red circles).

considered to represent curbs.

$$rowslines\ = \sum\nolimits_{i=1}^{r} with\ [p \le pixels\ \ per\ \ row \le q] \tag{3}$$

$$colslines\ = \sum\nolimits_{i=1}^{c} with\ [p \le pixels\ \ per\ \ col \le q] \tag{4}$$

$$rowspercent = \frac{rowslines}{r} \le \mu \tag{5}$$

$$colspercent = \frac{colslines}{c} \le \mu \tag{6}$$

The linear feature detection step removes large regions that do not represent curbs because their shape is not linear. Nevertheless, there could be short nonlinear regions that, due to their small size, satisfy the linear condition, even without representing a curb. To avoid that, the relationship between the number of columns and rows is determined for each region. The computed region aspect ratio must have a value between the established thresholds (m and n in Eq. (7)).

$$m \le \frac{\#cols}{\#rows} \le n \tag{7}$$

The final result of the linear feature detection is a binary image in which 1-value pixels represent the location of curbs in the studied area. Pixels with a DV = 0 represent the background. At this moment, curbs and roadsides are located planimetrically (x and y coordinates). Fig. 3 shows the results obtained in every step of the procedure for one slice of every study test site. The final binary images obtained for each site are shown in Fig. 3(e) and (f).

3.5. Edge detection

3.5.1. 3D cloud recovery

Once the curb binary image has been produced, it is necessary to recover the 3D point cloud to detect the upper and lower curb edges. Thus, we move from the 2D raster image to a new 3D point cloud. This new point cloud will be more manageable than the original one because it is formed only by those points of the rotated point cloud (obtained in Step 3.1), which are contained in the 1-value pixels of the binary image.

3.5.2. Edge point detection

From the cloud of candidate points, the upper and lower edges of curbs are detected by performing a +/−45° rotation around the x-axis, which is coincident with the average trajectory of the vehicle. Thus, the surfaces of the sidewalks, curbs, and roads, formerly horizontal and vertical, now have a slope of +/−45°, which is similar to the profile of a mountain and a valley (Fig. 4). This is valid even the road is sloped or flat due to the previous coordinate system change. This procedure helps in the edge curb extraction process because, in this way, the problem is reduced to detect changes in the slope sign.

In the data file provided by the laser scanner, the point cloud is organized according to the GPS time at which every point was scanned. To detect the points that represent the upper and lower edges of each curb, every point is compared with the previous and the next scanned points in the point cloud file. The slope of the vectors which links the studied point with its neighbor points is computed. If both calculated slopes have the same sign, the studied point represents a point in a vertical or horizontal surface (Fig. 5(a) and (c)). If the slopes have different signs, the studied point represents a change in slope and corresponds to an upper or lower curb edge (Fig. 5(b) and (d)).

The output of this step is a point cloud in which every point represents the upper or lower edges of the curbs that exist in the studied area (Fig. 6). From the edge curb point cloud, it is easy to determine those points in the scanned point cloud that belong to the sidewalk from those that belong to the road.

3.6. Estimation of non-detected boundaries

It is not always possible to detect the curbs and road boundaries based on MLS point clouds. Several obstacles on any given street may obstruct the view of curbs (street furniture, vehicles, or pedestrians that are between the MLS and the curb at the time of measurement). In other cases, the roadside exists but is not detectable with the proposed method. This occurs with access ramps at crosswalks: the curbs in these areas are at the same level as the road, making it difficult to detect the street boundary.

A method to estimate the locations of curbs and street boundaries is presented in this step. To perform it, we assume that non-detected boundaries must be aligned with their nearest neighboring boundaries correctly detected. From the 2D mask obtained in Step 3.4.3, a regional

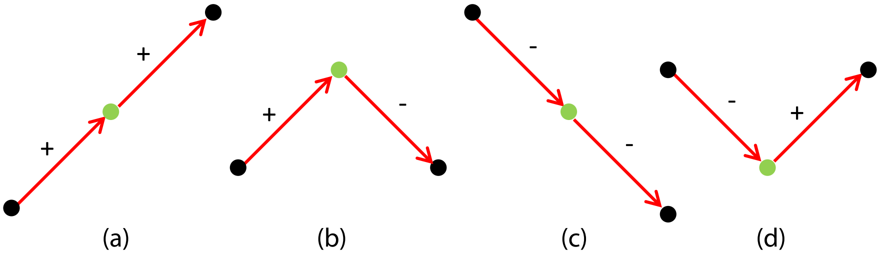


Fig. 5. Graphics (a) and (c) represent cases in which the studied point (the one in green color) does not represent a slope change; (b) and (d) show those situations in which the studied point represents a slope change and thus represents a curb edge.

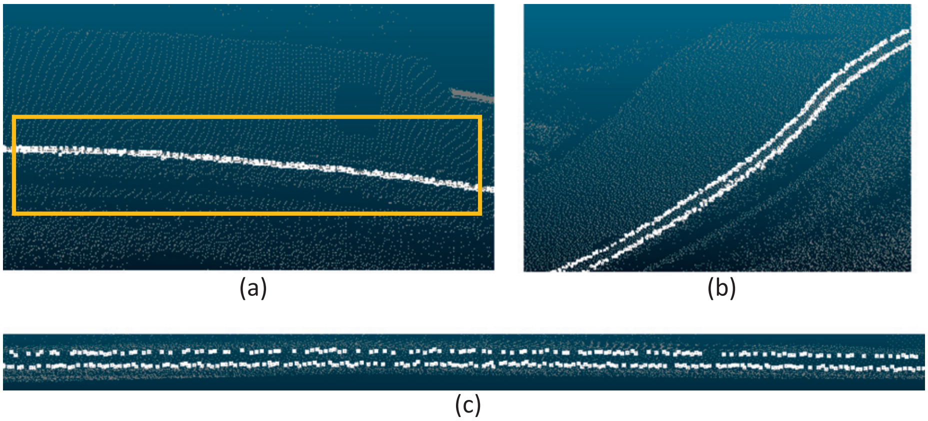


Fig. 6. (a) Curb section in the original point cloud, (b) curb section overlapped by detected curb edges (in white color), and (c) points detected as edges in a longitudinal profile marked in Fig. 7(a).

study is carried out in order to determine the direction of every curb region. Thus, for every region, three directions are computed: its direction, the nearest neighboring region's direction, and the direction of the vector that joins the centroids of both regions. These three two-dimensional vectors will determine the direction of both regions. If every vector has the same direction, the studied regions are aligned, and it is considered to be a roadside that was not detected by the proposed method. The cross-product is used to verify the fact that the vectors have the same direction. If the vectors are collinear, the cross-product between each pair must be null. In this case, we assumed that, between the two regions, a non-detected curb section exists (Fig. 7(a)). The proposed method cannot determine the existence of a non-detected boundary for those regions that are not collinear due to the fact that their directions are different (Fig. 7(b)).

The automatic method presented here for hidden boundaries curb estimation works properly only in straight curb sections. Other works, like those of Hervieu and Soheilian [36], have proposed a semi-automatic method for curb detection in curved sections, in which the user must supply the curb direction to the algorithm.

4. Test cases

4.1. MLS sensor

To determine the accuracy of the developed method, it was tested in two test sites acquired with the Lynx Mobile Mapper system, produced by Optech Inc. The lynx scanner collects survey-grade LIDAR data at 500,000 measurements per second with a 360° field of view (FOV). The Lynx also incorporates the POS LV 520, by Applanix, which integrates an IMU with a 2-antenna heading measurement system. LIDAR

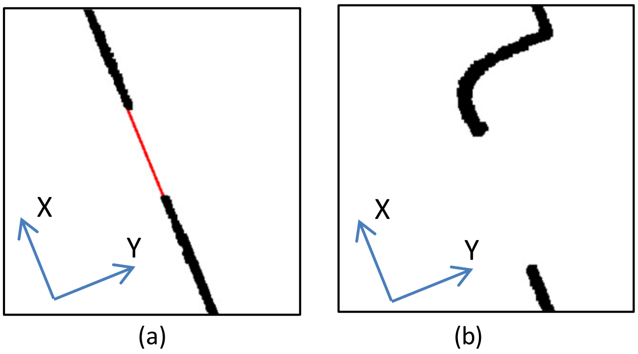


Fig. 7. (a) Collinear regions (in black) detected and the boundary estimated between both (in red); (b) non-collinear regions detected in the original point cloud with the proposed method.

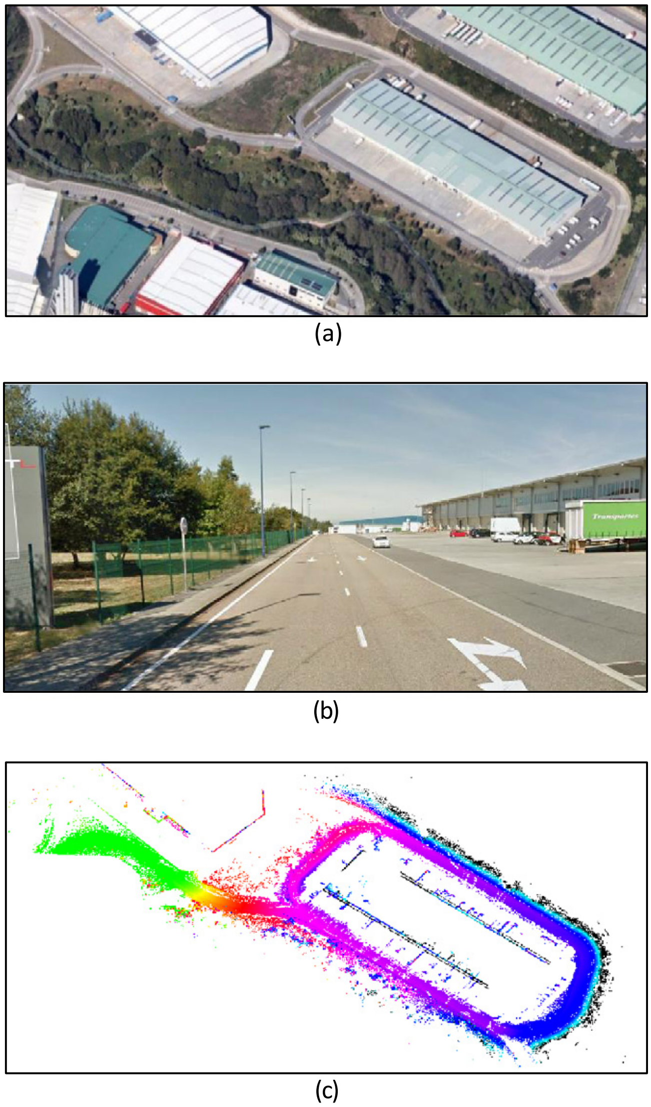


Fig. 8. Test Site 1: (a) aerial image, (b) street appearance and (c) measured point cloud.

sensors are located in the rear of a van. Each sensor registers points in a plane at 60° to the horizontal and 45° to the longitudinal axis of the vehicle (i.e. driving direction). This laser scanner provides absolute accuracies of 0.015° in heading, 0.005° in roll and pitch, 0.02 m in X, Y positions, and 0.05 m in Z. All these values are determined via differential GPS post-processing after data collection using GPS base station data [42]. One advantage provided by the MLS is the possibility of generating a geo-referenced point cloud combining LIDAR sensor data with IMU and a GPS installed in the mobile platform. The spatial resolution of the point cloud depends on the scan frequency of the LIDAR heads, the pulse repetition rate, the speed of the vehicle, and the distance to the measured objects [43].

4.2. Algorithm settings

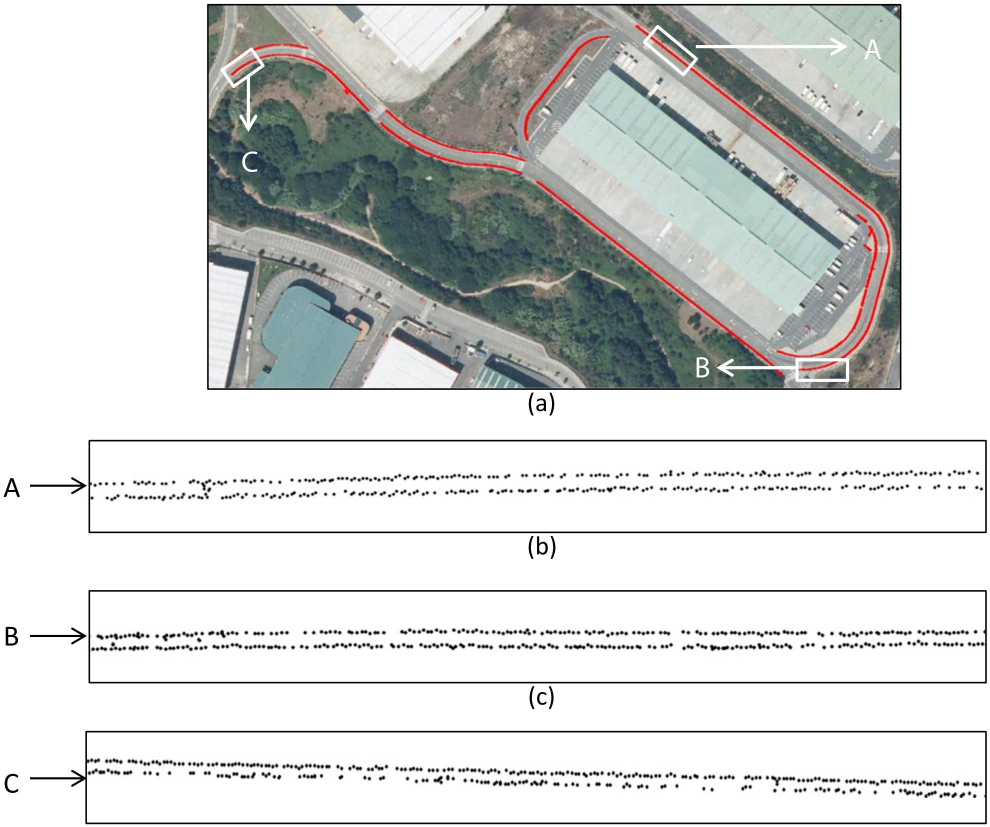
The accuracy of the curb detection is directly related to the correct selection of different parameters. These settings depend on the attributes of the input point cloud and the environment of the studied road. We have realized that the critical parameters are (1) the pixel size of the grid, (2) the height difference restrictions in the threshold step, and (3) the point density. For the test sites used in this work, the pixel size was established at 5 × 5 cm. The characteristics of the existing curbs directly affect the height difference (Δh) parameter. Typically, the curb height is about 0.15 m, but it can vary depending on the studied area. The minimum (Hmin) and maximum (Hmax) curb height thresholds for both test sites were set to 0.10 m and 0.20 m, respectively. The point density (Dmin) in each pixel was set to 5 points/pixel. Other parameters, such as line width, line percent, and region aspect ratio, are less dependent on the characteristics of the cloud. They take similar values in every case because they refer to the properties of lines rather than those of the curbs. The line width (p and q thresholds) was set between three and six pixels, and the line percent (μ) was set to 80%. The region aspect ratio was fixed at 3 to 1. The values of the parameters for the study cases are listed in Table 1. The same settings were applied to both test sites. In both test sites the cloud is splited in 150 m long slices. Every slice is computed separately in order to decrease the computational cost of the procedure. This split makes possible to detect curbs in curved sections.

4.3. Test Site 1

The point cloud used in Test Site 1 was measured in a polygonal park close to Vigo, in the north of Spain. In this area, there is an industrial-type building with an area of 9000 square meters. Around it, there is a two-way road that is 800 m long with both, straights and curved sections (Fig. 8). The presence of cars, trucks, and other obstacles, such as fences, lampposts, and poles, makes the roadside detection more difficult. The detected point cloud consists of more than 45 million points. This test site was used to check the capacity of the proposed method to detect curb edges.

4.3.1. Reference data

To evaluate the accuracy of the curb detection, a manual extraction of the road boundaries from the original point cloud was carried out. It was performed by digitizing the observed road borders from the point cloud as the ground truth data. For Test Site 1, the ground truth has a length of almost 1098 m. The evaluation of the results was carried out by comparing the curbs extracted via the proposed method with the previously prepared ground truth. This was performed by using three indices commonly used in the evaluation of



(a)

(b)

(c)

(d)

Fig. 9. (a) The point cloud formed by the upper and lower edges detected (in red), overlapping an aerial image of Test Site 1; (b), (c) and (d) represent points detected as upper and lower edges in A, B, and C details.

Table 2
Accuracy of the detection method at Test Site 1.

Test Site 1	Algorithm-detected (AD)	User-detected	FP	FN	TP = AD − FP
Data-present curbs	1072.5	1097.8	18.1	25.3	1054.4
Completeness	97.65%	Correctness	98.31%	Quality	96.05%

road detection: completeness (Eq. (8)), correctness (Eq. (9)), and quality (Eq. (10)) [11,44,45].

$$Completeness = \frac{length\ \ of\ \ matched\ \ reference}{length\ \ of\ \ reference} = \frac{TP}{TP + FN} \tag{8}$$

$$Correctness = \frac{length\ \ of\ \ matched\ \ extraction}{length\ \ of\ \ extraction} = \frac{TP}{TP + FP} \tag{9}$$

$$Quality = \frac{length\ \ of\ \ matched\ \ extraction}{length\ \ of\ \ extracted + unmatched\ \ reference} = \frac{TP}{TP + FP + FN} \tag{10}$$

where TP (true positive) represents the length of the curb detected that matches the reference roadside, FP (false positive) represents the length of the detected curbs that do not matching with the ground truth, and FN (false negative) represents the total length of the undetected curbs that exist in the ground truth.

4.3.2. Results

For a visual analysis of the results obtained in Test Site 1, the detected curbs have been superimposed on an ortho-image of the studied area (Fig. 9(a)). The result of the curb edge detection method is shown in three detailed images, which represent the curved and straights section of the test site (Fig. 9(b), (c) and (d)).

The proposed method identified 1072.5 m as curb from the Test Site 1 data, of which 18 m represents false positives caused by low vegetation and elements with geometry similar to those of curbs, such as car bottoms and steps. 1054.5 m of the detected curb matched the ground truth curb, and 25.3 m that belonged to the ground truth were not detected via the method due to the occlusion of the curb by grass or low vegetation. In these cases it is not possible to achieve the detection because of the change in the curb geometry. The parameters used to measure the accuracy of our method are written in Table 2. It can be appreciated that they are all above 96%.

4.4. Test Site 2

Test Site 2 corresponds to a 250 meter section of Rua Progreso Street in Ourense, a city in the north of Spain. This is a typical urban area that has a road with structured road boundaries in the form of curbs and ramps at crosswalks and garages (Fig. 10). These elements and others, such as cars and pedestrians, create shadows in the point cloud. Test Site 2 was used to test the performance of the proposed method of estimating those non-detected boundaries. The point cloud corresponding to Test Site 2 consisted of more than 18 million points.

4.4.1. Reference data

The results obtained for the detection and estimation procedures at Test Site 2 were evaluated by comparing the extraction with the two ground truth datasets. One ground truth was composed by curbs observed in the point cloud and it consists of more than 370 m of curbs. The other ground truth was formed by occluded curbs and

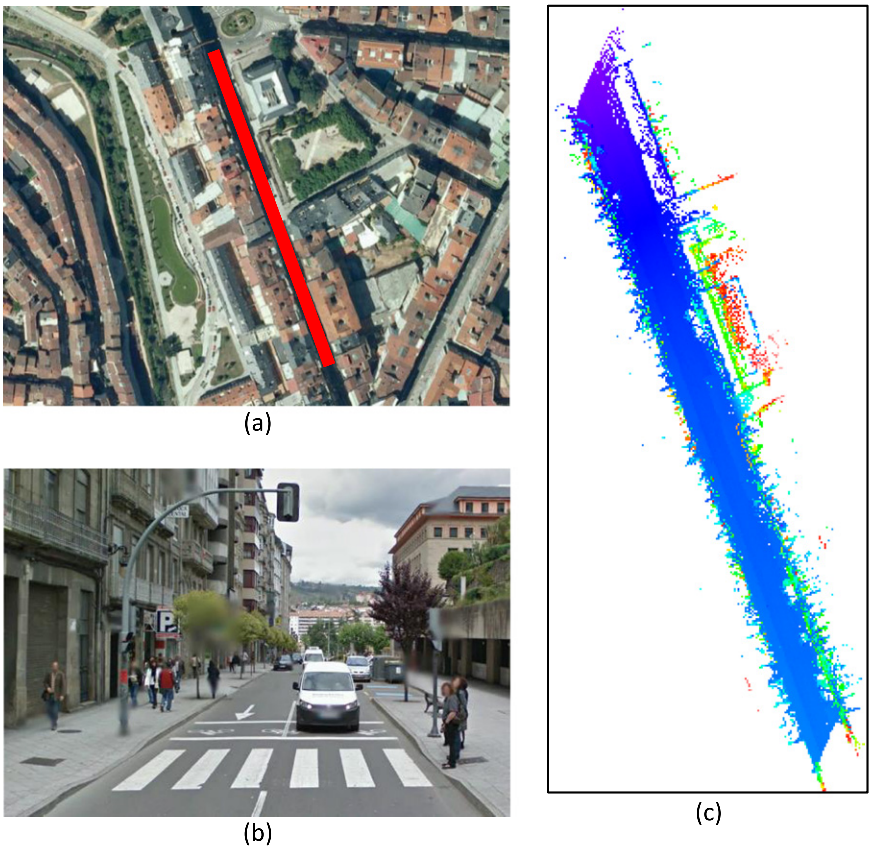


Fig. 10. Test site 2: (a) aerial image, (b) street appearance, and (c) measured point cloud.

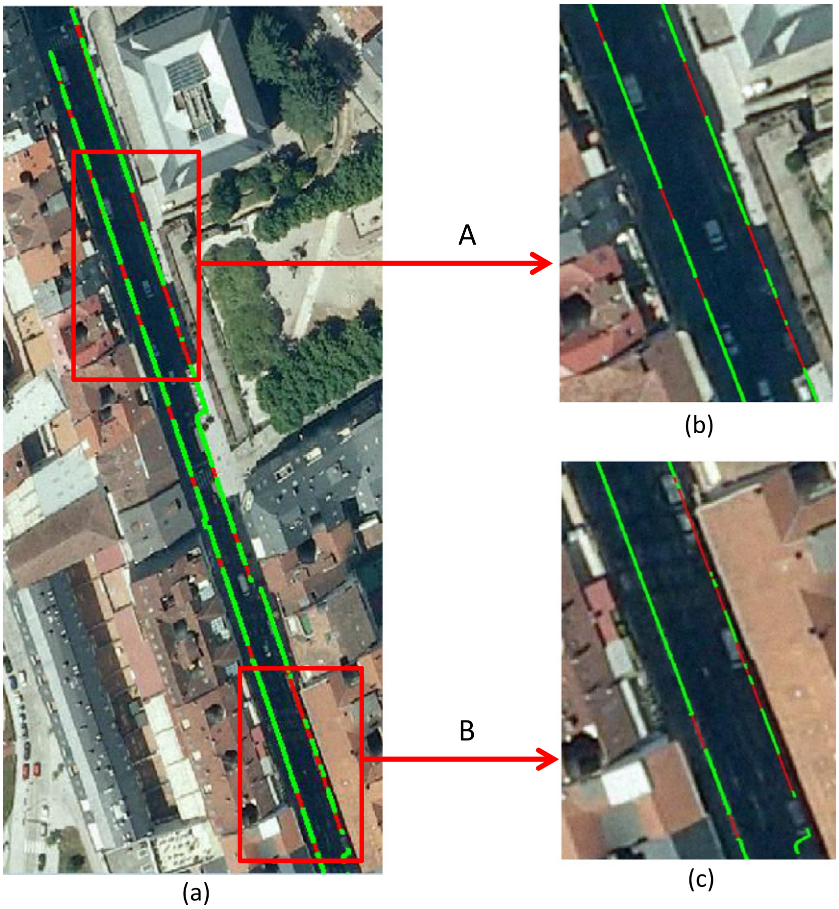


Fig. 11. (a) The detected curbs (green color) and estimated boundaries (red color) for Test Site 2; (b) and (c) show the results obtained in details A and B.

ramps at sidewalks and garages. It consisted of more than 115 m and was used to evaluate the results of the curb estimation method.

4.4.2. Results

The detected curbs (green lines) and estimated sections (red lines) were rasterized and superimposed on an ortho-image of the studied area (see Fig. 11) in order to improve the visualization. Most of the detected segments in Fig. 11 are parallel to the vehicle trajectory, although a segment perpendicular to that is shown in the lower right corner of the image (c).

According to the results listed in Table 3, almost 358 m of curb was detected using our algorithm, of which only 2 m did not match the data present in the ground truth curbs. However, around 15 m that existed in the ground truth were not detected. According to these results, completeness, correctness and quality obtained were 95.9%, 99.36%, and 95.31%, respectively. Regarding the hidden boundaries, 112 m of the 116 m of hidden boundaries were delineated. Almost 4 m of hidden boundaries in the ground truth were not estimated via the method. These FN are located in curved sections. There was no FP in the estimation procedure, meaning that the completeness and quality values took the same values, providing a correctness of 100%. Almost 97% completeness and quality were achieved for the boundary estimation method.

Table 3
Accuracy of the detection and estimation method at Test Site 2.

Test Site 2	Algorithm detected	User detected	FP	FN	TP = AD − FP
Data-present curbs	357.5	370.4	2.3	15.2	355.2
Data-hidden boundary	111.8	115.5	0	3.7	111.8

Test Site 2	Data-present Curbs	Data-hidden boundaries
Completeness	95.9%	96.8%
Correctness	99.36%	100%
Quality	95.31%	96.8%

5. Conclusions

In this paper, a novel method to detect and estimate curbs and street boundaries from MLS data is presented. The method begins with a coordinate system transformation and a rasterization that simplify the calculations involved and allow the use of image processing techniques to define linear features. Finally, edge points are detected by rotating the point cloud and looking for those points that represent a slope change. The upper and lower edges of the curb are extracted, resulting in a 3D representation of the road boundaries. In addition, an estimation of the boundaries is carried out in occluded areas located between straight segments by means of the direction and distance analysis of the neighboring detected segments.

The method was tested in two areas, one of which corresponded to an urban environment and the other corresponded to an industrial environment. The results obtained show completeness, correctness, and quality values higher than 95% for both detected and estimated boundaries. These results are quite better than the ones provided by other methods as [35] and [46] and are accurate enough for an autonomous driving system. From the results obtained at both test sites, it is possible to conclude that the proposed method is valid (1) for detecting the change of shapes road boundaries both in straight and curved road sections, (2) for accurately estimating the location of occluded curbs and undetected straight boundaries, and (3) for accurately determining the upper and lower curbs' edges. However, it is still difficult to deal with occluded curbs in curved sections and with boundaries without 3D shapes. In the near future, other variables as the topology or the texture will be incorporated to enhance the estimation method.

Acknowledgments

The authors would like to thank the Spanish Ministry of Science and Innovation for financial support, Project No. CGL2010-15357/BTE and Project No. BIA2011-26915.

References

[1] X. Ding, W. Kang, J. Cui, L. Ao, Automatic extraction of road network from aerial images, International Symposium on Systems and Control in Aerospace and Astronautics (ISSCAA), 2006, pp. 220–223.

[2] R. Schmidt, H. Weisser, P. Schulenberg, H. Goellinger, Autonomous driving on vehicle test tracks: overview, implementation and results, Proceedings of the IV IEEE Intelligent Vehicles Symposium, 2000, pp. 152–155.

[3] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M.N. Clark, H. Bae, T. Brown, D. Demitrish, J. Struble, M. Darms, Autonomous driving in urban environments: boss and the urban challenge, J. Field Rob. 25 (2008) 425–466.

[4] J. Maye, R. Kaestner, R. Siegwart, Curb detection for a pedestrian robot in urban environments, IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 367–373.

[5] J.B. Mena, State of the art on automatic road extraction for GIS update: a novel classification, Pattern Recogn. Lett. 24 (2003) 3037–3058.

[6] M.F. Auclair-Fortier, D. Ziou, C. Armenakis, S. Wang, Survey of work on road extraction in aerial and satellite images, Can. J. Remote. Sens. 27 (2001) 76–89.

[7] C. Zhang, Towards an operational system for automated updating of road databases by integration of imagery and geodata, ISPRS J. Photogramm. Remote Sens. 58 (2004) 166–186.

[8] S. Clode, F. Rottensteiner, P. Kootsookos, E. Zelniker, Detection and vectorization of Roads from Lidar Data, Photogramm. Eng. Remote Sens. 73 (2007) 517–535.

[9] J. Amini, M.R. Saradjian, J.A.R. Blais, C. Lucas, A. Azizi, Automatic road-side extraction from large scale imagemaps, Int. J. Appl. Earth Obs. Geoinf. 4 (2002) 95–107.

[10] M. Song, D. Civco, Road extraction using SVM and image segmentation, Photogramm. Eng. Remote Sens. 70 (2004) 1365–1371.

[11] S. Clode, P.J. Kootsookos, F. Rottensteiner, The Automatic Extraction of Roads from Lidar Data, presented at the ISPRS, Istanbul, Turkey, 2004.

[12] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, A. Baumgartner, Automatic extraction of roads from aerial images based on scale space and snakes, Mach. Vis. Appl. 12 (2000) 23–31.

[13] X.Y. Jian, H. Bunke, Fast segmentation of range images into planar regions by scan line grouping, Mach. Vis. Appl. 7 (2) (1994) 115–122.

[14] M. Varela-González, H. González-Jorge, B. Riverio, P. Arias, Performance Testing of 3D Point Cloud Software, presented at the ISPRS Workshop Laser Scanning, Antalya, Turkey, 2013.

[15] D. Belton, K. Bae, Automating post-processing of terrestrial laser scanning point clouds for road feature surveys, Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci. XXXVIII (2010) 74–79.

[16] D. Gonzalez-Aguilera, A.L. Muoz, J.G. Lahoz, J.S. Herrero, M.S. Corchon, E. Garcia, Recording and Modeling Paleolithic Caves through Laser Scanning, International Conference on Advanced Geographic Information Systems & Web Services (GEOWS), 2009, pp. 19–26.

[17] X.-j. Cheng, H.-f. Zhang, R. Xie, Study on 3D laser scanning modeling method for Large-Scale history building, International Conference on Computer Application and System Modeling (ICCASM), 2010 pp. V7-573-V7-577.

[18] D. Tapete, G. Gigli, F. Mugnai, P. Vannocci, E. Pecchioni, S. Morelli, R. Fanti, N. Casagli, Correlation between erosion patterns and rockfall hazard susceptibility in hilltop fortifications by terrestrial laser scanning and diagnostic investigations, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2012, pp. 4809–4812.

[19] R. Argüelles-Fraga, C. Ordóñez, S. García-Cortés, J. Roca-Pardiñas, Measurement planning for circular cross-section tunnels using terrestrial laser scanning, Autom. Constr. 31 (2013) 1–9.

[20] M. Pejić, Design and optimisation of laser scanning for tunnels geometry inspection, Tunn. Undergr. Space Technol. 37 (2013) 199–206.

[21] H. Park, S. Lim, J. Trinder, R. Turner, 3D surface reconstruction of Terrestrial Laser Scanner data for forestry, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2010, pp. 4366–4369.

[22] W. Ni, G. Sun, Z. Guo, Y. Pang, Characterization of soil surface roughness from terrestrial laser scanner data, IEEE International Geoscience and Remote Sensing Symposium, 2009, pp. II-428–II-431.

[23] C. Perez-Gutierrez, J. Martinez-Fernandez, N. Sanchez, J. Alvarez-Mozos, Modeling of soil roughness using terrestrial laser scanner for soil moisture retrieval, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2007, pp. 1877–1880.

[24] G. Vosselman, Advanced Point Cloud Processing, presented at the Photogrammetric week, Wichmann, 2009.

[25] M. Rutzinger, S. Oude Elberink, S. Pu, G. Vosselman, Automatic extraction of vertical walls from mobile and airborne laser scanning data, presented at the The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Paris, 2009.

[26] K. Hammoudi, F. Dornaika, N. Paparoditis, Extracting building footprints from 3D point cloud using terrestrial laser scanning at street level, presented at the ISPRS/CMRT09, Paris, 2009.

[27] C. Brenner, Extraction of Features from Mobile Laser Scanning Data for Future Driver Assistance Systems, in: M. Sester, L. Bernard, V. Paelke (Eds.), Advances in GIScience, Springer Berlin Heidelberg, 2009, pp. 25–42.

[28] M. Lehtomäki, A. Jaakkola, J. Hyyppä, A. Kukko, H. Kaartinen, Detection of vertical pole-like objects in a road environment using vehicle-based laser scanning data, Remote Sens. 2 (2010) 641–664.

[29] J. Ruyi, K. Reinhard, V. Tobi, W. Shigang, Lane detection and tracking using a new lane model and distance transform, Mach. Vis. Appl. 22 (2011) 721–737.

[30] R. Labayrade, J. Douret, D. Aubert, A multi-model lane detector that handles road singularities, IEEE Intelligent Transportation Systems Conference (ITSC '06), 2006, pp. 1143–1148.

[31] J. Siegemund, D. Pfeiffer, U. Franke, W. Forstner, Curb reconstruction using Conditional Random Fields, IEEE Intelligent Vehicles Symposium, 2010, pp. 203–210.

[32] K. ZuWhan, Robust lane detection and tracking in challenging scenarios, IEEE Trans. Intell. Transp. Syst. 9 (2008) 16–26.

[33] G. Zhao, J. Yuan, Curb detection and tracking using 3D-LIDAR scanner, IEEE International Conference on Image Processing (ICIP), 2012, pp. 437–440.

[34] T. Weiss, K. Dietmayer, Automatic Detection of Traffic Infrastructure Objects for the Rapid Generation of Detailed Digital Maps using Laser Scanners, IEEE Intelligent Vehicles Symposium, 2007, pp. 1271–1277.

[35] B. Yang, L. Fang, J. Li, Semi-automated extraction and delineation of 3D roads of street scene from mobile laser scanning point clouds, ISPRS J. Photogramm. Remote Sens. 79 (2013) 80–93.

[36] A. Hervieu, B. Soheilian, Semi-automatic road/pavement modeling using mobile laser scanning, presented at the City Models, Roads and Traffic—CMRT13, Antalya, Turkey, 2013.

[37] P. Kumar, C.P. McElhinney, P. Lewis, T. McCarthy, An automated algorithm for extracting road edges from terrestrial mobile LiDAR data, ISPRS J. Photogramm. Remote Sens. 85 (2013) 44–55.

[38] VirtualGrid, Available: http://vrmesh.com, 2014.

[39] N. Haala, C. Brenner, Extraction of buildings and trees in urban environments, ISPRS J. Photogramm. Remote Sens. 54 (1999) 130–137.

[40] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*: Pearson Education, 2011.

[41] B. Rodriguez-Cuenca, J.A. Malpica, M.C. Alonso, Region-growing segmentation of multispectral high-resolution space images with open software, Proceedings of IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2012, pp. 4311–4314.

[42] I. Puente, H. González-Jorge, B. Riveiro, P. Arias, Accuracy verification of the Lynx Mobile Mapper system, Opt. Laser Technol. 45 (2013) 578–586.

[43] C. Cabo, C. Ordoñez, S. García-Cortés, J. Martínez, An algorithm for automatic detection of pole-like street furniture objects from Mobile Laser Scanner point clouds, ISPRS J. Photogramm. Remote Sens. 87 (2014) 47–56.

[44] C. Heipke, H. Mayer, C. Wiedemann, O. Jamet, Evaluation of automatic road extraction, Int. Arch. Photogramm. Remote. Sens. 32 (3-2W3) (1997) 47–56.

[45] J. Hu, A. Razdan, J.C. Femiani, C. Ming, P. Wonka, Road network extraction and intersection detection from aerial images by tracking road footprints, IEEE Trans. Geosci. Remote Sens. 45 (2007) 4144–4157.

[46] L. Zhou, G. Vosselman, Mapping curbstones in airborne and mobile laser scanning data, Int. J. Appl. Earth Obs. Geoinf. 18 (2012) 293–304.