Service Oriented Computing and Applications
https://doi.org/10.1007/s11761-024-00399-7

SPECIAL ISSUE PAPER

Detecting and tracking a road-drivable area with three-dimensional point clouds and IoT for autonomous applications

Mahmood H. Enad[1] · Omar I. Dallal Bashi[2] · Shymaa Mohammed Jameel[3] · Asaad A. Alhasoon[4] · Yasir Mahmood Al Kubaisi[5] · Husamuldeen K. Hameed[6]

Received: 19 February 2024 / Revised: 29 March 2024 / Accepted: 6 April 2024
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2024

Abstract

This work presents a Light Detection and Ranging (LiDAR)-based point cloud method for detecting and tracking road edges. Initially, this work explores the progress in detecting road curb issues. A dataset (called PandaSet) with a Pandar64 sensor to capture different city scenes is used. LiDAR point cloud, as part of an IoT ecosystem, detects the road curb and requires distinguishing the right and left road curbs with regard to the ego car. The curb point's features use Random Sample Consensus (RANSAC)-based polynomial quadratic approximation to obtain the prospect curb points to eliminate false positive ones. Through extensive experiments, we demonstrate the effectiveness and reliability of our method under various traffic and environmental conditions. Our results showcase a maximum drift of 1.62 m for left curb points and 0.87 m for right curb points, highlighting the superior accuracy and stability of our approach. This LiDAR-based curb detection framework paves the way for enhanced lane recognition and path planning in autonomous driving applications.

Keywords LiDAR sensor · Point cloud processing · Random sample consensus algorithm · Polynomial quadratic approximation · Feature extraction

1 Introduction

Autonomous vehicles rely heavily on accurate perception of their surroundings, and road edge detection plays a crucial role in safe navigation. LiDAR-based methods have emerged as a promising approach due to their high precision and ability to operate in diverse lighting conditions [1, 2]. Several challenges have been assigned in previous studies with curb detection. These include (1) accurately differentiating between the left and right side of the road is crucial for lane-level navigation and maneuvering [3], (2) variations in weather, lighting, and road markings can make curb detection challenging [4], (3) clutter from vegetation, obstacles, and road imperfections can lead to misinterpretations [5].

Traditional methods often rely on intensity thresholds and geometric features of point clouds. However, these can struggle with varying environmental conditions and complex urban scenes. Deep learning and point cloud processing techniques are gaining traction, offering improved accuracy and robustness [6]. These methods extract intricate features from LiDAR data, enhancing curb detection even in challenging situations. Precise curb detection serves as the foundation for various autonomous driving tasks, including

Mahmood H. Enad and Asaad A. Alhasoon have contributed equally.

✉ Mahmood H. Enad
mahmood.enad@atu.edu.iq

✉ Asaad A. Alhasoon
ahs4771384@gmail.com; asaad.alhasoon@gmail.com

Shymaa Mohammed Jameel
shymaa792003@yahoo.com

1 Department of Electrical Techniques, Technical Institute of Karbala, Al-Furat Al-Awsat Technical University, Najaf, Iraq

2 Medical Technical Institute, Northern Technical University, Mosul, Iraq

3 Iraqi Commission for Computers and Informatics, Baghdad 10009, Iraq

4 Ministry of Scientific and Technology Research, Jadrriyah, Baghdad, Iraq

5 Department of Sustainability Management, Dubai Academic Health Corporation, Dubai, UAE

6 Department of Telecommunications and Information, High Institute of Telecommunications and Post, Baghdad, Iraq

Published online: 09 May 2024

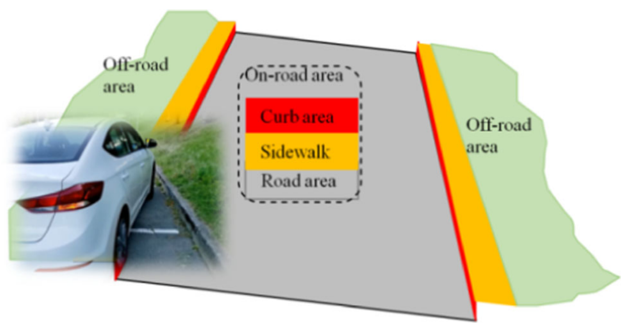


Fig. 1 A photo showing the road curb

lane recognition, obstacle avoidance, and localization and mapping. Research in LiDAR-based curb detection continues to evolve, focusing on integrating additional sensor data that can further enhance accuracy and robustness, real-time performance optimization to ensure fast and efficient processing, and adaptability to diverse scenarios that can handle unique road designs and environments [7].

Technology for autonomous driving is developing quickly to satisfy the demands of highway transportation and safety effectiveness. Road boundary identification is not an easy task because of the problems like the small size targets, shadows, and occlusions. Although there are obstructions in the road, curb identification algorithms are highly accurate and resilient under both obscured and uneven road curb conditions [8]. Photographs showing road definition curbs are shown in Fig. 1.

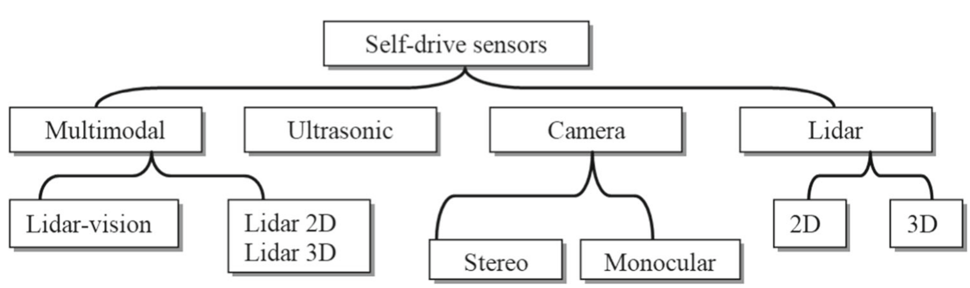
Numerous previous techniques addressed road curb detection issues. 3D point cloud with LiDAR sensor is a robust method to solve these problems, especially under curved road scenarios, obstacle occlusions, and road discontinuities. Merging advanced vision-based detection methods with 3D geometric reasoning can estimate a curb distance of more than 90% accuracy in real-time [9]. Yang et al. also proposed road curb detection based on laser-based 3D point clouds [10] using binary kernel descriptor (BKD) as a 3D local feature for extracting road data. A 3D point cloud LiDAR sensor and VeCaN Tongji University dataset in real-time curb recognition was also proposed by [11]. The average processing time for each frame to around 12 ms, while the mean harmonic, precision, and average recall, are all above 80% [12]. The majority of the road surface points are deleted by evaluating the horizontal and vertical continuity between points in the same laser beam after the points outside the road regions are first removed using the Random Sample Consensus (RANSAC) algorithm [13–16]. Many approaches have been developed over the years and can be categorized according to a number of factors, including the type of sensor used to collect the data, as illustrated in Fig. 2.

The above-aforementioned studies discussed many road curb detection algorithms by creating intelligent vehicles, where accurate and speedy road curb detection is essential. However, curb recognition requires involving more features from the environment to accurately estimate the curb points. Therefore, this work contributes to a LiDAR-based point cloud strategy for detecting and tracking road edges, where the curb point's features use RANSAC-based polynomial quadratic approximation to obtain the applied curb points to eliminate false positive ones. The curb points are identified and segmented from the on-road point cloud by performing the features of; 1) the vertical and horizontal continuity according to its immediate neighboring points, 2) the height difference to check the maximum difference and the standard deviation of the height near a point, and 3) the smoothness of the area near a point, where a plane point is that point with lower smoothness values, and an edge point is that point with higher values.

2 Methods and materials

An available publicly dataset (PandaSet) with Pandar64 sensor to capture different city scenes, which contains Lidar scans of point clouds, is used [17]. Lidar point cloud detects the road curb and requires distinguishing the right and left road curbs with regard to the ego car. The Lidar sensors are positioned on the vehicle ceiling. The considered dataset contains 50 preprocessed organized point clouds 64-by-1856-by-3 array of PCD format each. It includes 13 classes of ground truth data in PNG format and semantic segmentation labels. MATLAB tools have been used to conduct the developed model. From the Lidar sensor data and the captured point clouds, the detection process of road curbs includes:

Fig. 2 Types of self-drive sensors



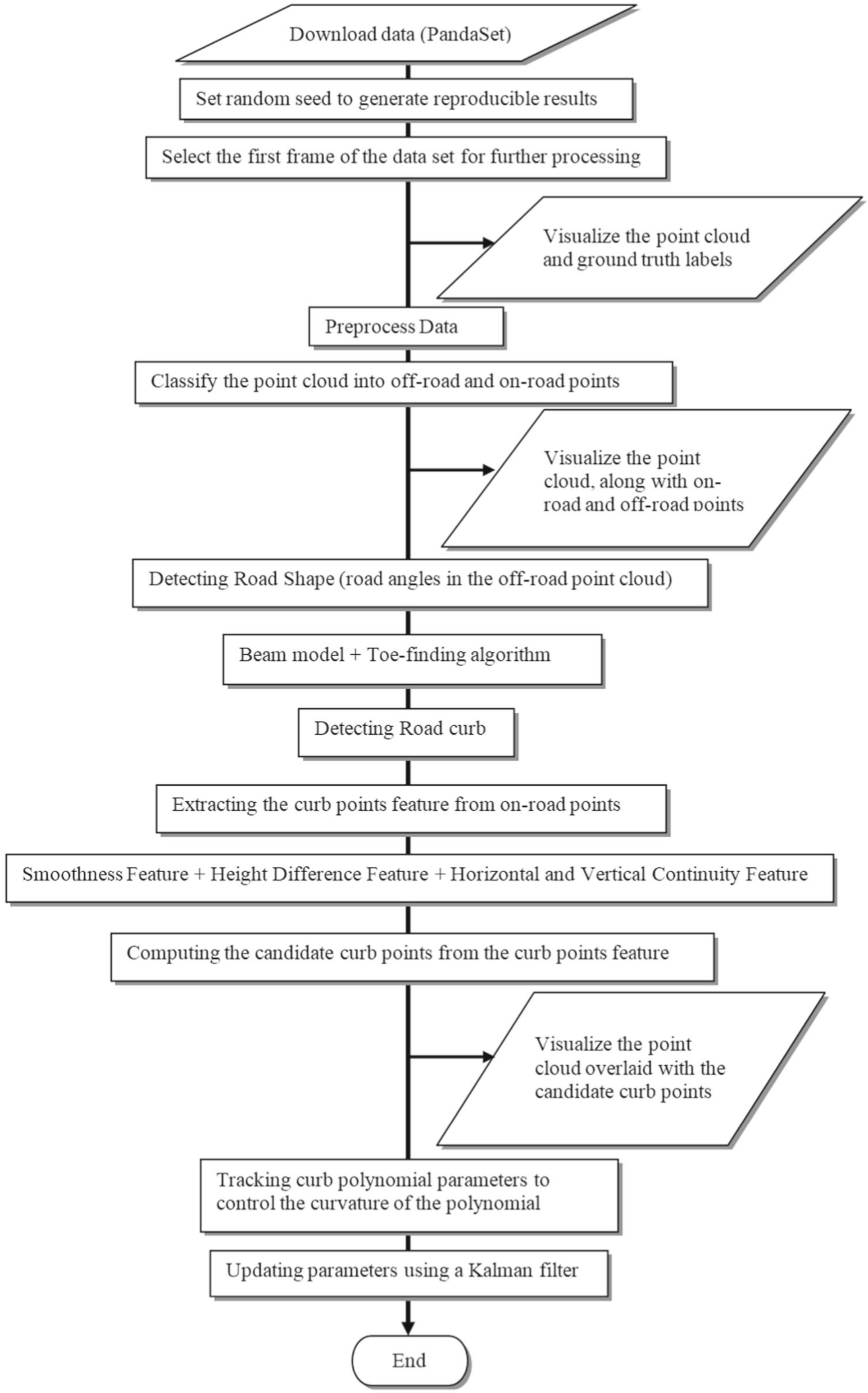


Fig. 3 The developed diagram of the Lidar-based data with 3-D point clouds to detect road-drivable areas for autonomous driving

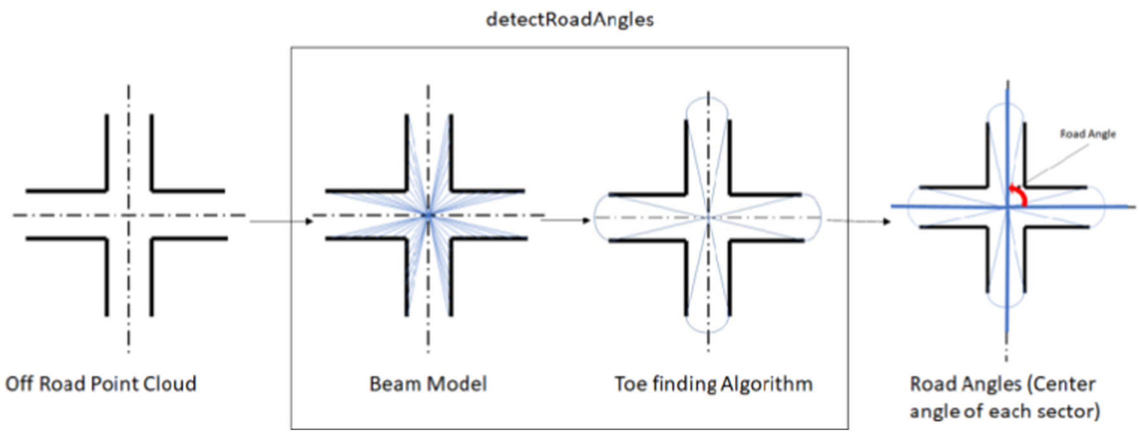


Fig. 4 A graphical representation of detecting road shape stages for the off-road point cloud. By using the on-road point cloud, the segmenting and computing of the road curb point are performed. This includes extracting and classifying the curb points feature by following the spatial features to model the road curbs

- Extraction of an ROI (region of interest).
- Classifying off-road and on-road points.
- Using the off-road points to recognize road angles.
- Using the on-road points to identify candidate curb points.

2.1 Data Preprocessing

Initially, we identify an ROI from the point clouds and categorize the points inside it as off-road or on-road points as a pre-handing step for finding the curb line. Beyond a certain distance, the point cloud data is sparse due to the installed position of the Lidar sensor. According to [18], the vision-based elevation accuracy for a specified depth Z relies on different parameters, such as disparity uncertainty and focal distance F, and baseline B given by:

$$Z_e = \frac{Z^2 D_e}{BF - D_e Z} \tag{1}$$

By defining an ROI that is only a certain distance from the sensor, the point cloud is taken into account to extract dense enough for additional handing out. The flowchart of the developed approach is shown in Fig. 3.

The point cloud has been classified into off-road and on-road points with the labels vector represented by: [Buildings, Signs, Road Barrier, Pedestrian, Other Vehicle, Truck, Car, Side Walk, Road Marking, Road, Ground, Vegetation, and Unlabeled]. The off-road data point contains some objects and buildings. The on-road points include sidewalks, roads, and ground. Visualizing point clouds over off-road and on-road points.

2.2 Road shape detection

The process of 3-D Point cloud models of the environment induced by Lidar sensors provides an output for the vehicle's tracking control and object segmentation system, which can then conduct the necessary movements for independent driving. This includes steering, acceleration, and braking control actions, where Kalman filters are adopted to track the position and velocity of detected objects over time. A local map is generated using the point cloud data to determine the vehicle's location. Object detection data association from different frames is used to maintain a consistent track of each object on the map. The path angles in the off-path point cloud have been identified, while the beam model is applied to the off-road points as detailed in [11] and [19]. The road angles are obtained by a modified toe-finding algorithm to the normalizing beam lengths [20].

A graphical representation of the off-road point cloud, the adopted road angle detection methods including the Beam model and Toe-finding algorithm, to produce road angles (center angle of all sectors) is shown in Fig. 4.

2.3 Road curbs detection

From the on-road point cloud, we employ a function called "*segmentCurbPoints*" for segmenting and computing the road curb point, which performs the following steps:

(1) From the on-path data point, we extract and classify the curb points feature by following the spatial features to model the road curbs. These features include; Smoothness Feature [19], which examines the area smoothness close to a point. A lower value means that the point is a plane pinpoint, and a higher smoothness value means an edge point. Height dissimilarity characteristic [19], this element examines the maximum difference and the

standard deviation of the altitude in the region of a point. Third, the Vertical and Horizontal Continuity Attribute [11], these characteristics inspect the vertical and horizontal continuity of a point according to its nearest adjacent point. The feature curb points are the points that meet the attribute criteria for the aforementioned characteristics.

(2) From the mark curb points, we calculate the applicant curb points. False positives could be present in the feature curb points. The function extra examines the curb point's features using RANSAC-based polynomial quadratic approximation to obtain the applicant curb points in order to eliminate the false positives.

2.4 RANSAC algorithm

Figure 5 represents a diagram visualizing the key technique used in curb detection with 3D lidar point clouds, which is the RANSAC algorithm. This method identifies the most possible curb model from cluttered point or potentially noisy cloud data. It runs with random selecting iterations for minimal point sets to fit a curb model to these points, and therefore evaluate the model fitting for the outstanding data.

The diagram sequence demonstrates how point's likely belonging to the curb (inliers) and points not part of the curb (outliers) are recognized according to distance thresholds. Lastly, the model with the most inliers is chosen after a set number of iterations as the most feasible representation of the curb. This technique assists extracting an accurate and clean curb model from the raw Lidar point cloud information.

2.5 Curb points tracking

In the tracking curb points stage, looping via the Lidar and process data to track and extract the prospect curb points. The reliability of curb detection is increased by tracking curb points. Curb tracking can be stopped at segmented roadways and resumed when the ego vehicle travels off of those roads. The tracking of curb points applies to fit with a polynomial model on XY-data by a 2-D polynomial expressed by $y = ax2 + bx + c$, such that the parameters a, b, and c represent the polynomial values. Curb detecting following includes a two-stage technique:

- To manage the drift of the polynomial, tracking of "c" polynomial parameter of the curb point.
- To control the polynomial curvature, the tracking of curb polynomial parameters a and b is performed.

The updating process for these parameters is implemented by a constant velocity motion model of the Kalman filter and the tracking of curb points is demonstrated in Algorithm 1.

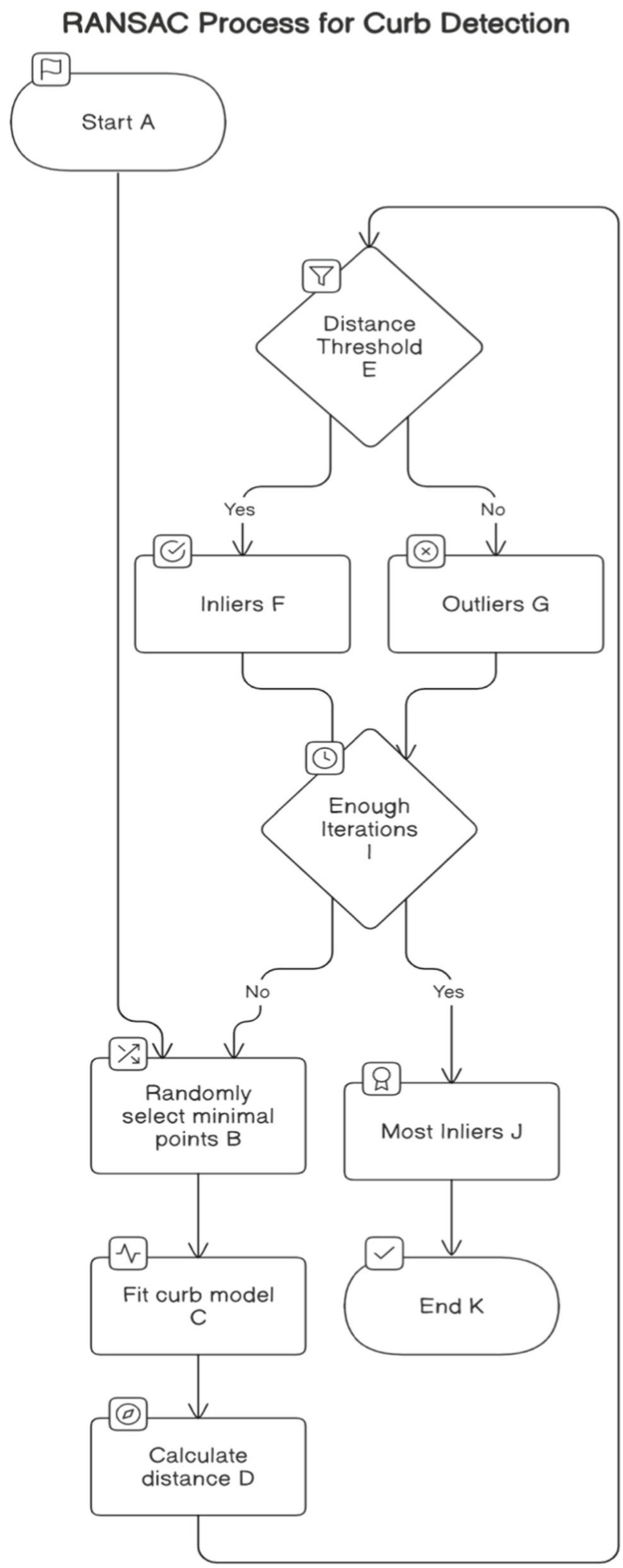


Fig. 5 RANSAC algorithm for curb detection

```
%Initializing Kalman filter, Finding ROI, and the XY points lying in
Start loop for count =point cloud data
%input ground truth labels, Checking curb tracking status, Computing %No. of road segments behind
and in front of the ego vehicle, calculate %road angles using off-road data, and classify off-road/on-road
points.
  if there is Tracking
    If No Front < No. Rear Segments OR No. of (road Ang) = 2;          Tracking = true;
    end
  else
    If No. Front Road Segments > No. Rear Road Segments
      Track= false;   Left Curb Track = false; Right Curb Track = false;
    End
  End
%Transform candidate points from previous frames to current frame
  If prev. Off-Road point cloud = empty
    Grid Step = 0.5; Prev. Cand Curb Points = pc transform;
  End
%Compute the curb points using the on-road points
  if RMSE > 0.7
     Road Ang, Horizontal Angular Resolution = 0.24;
  Else
    Cand Curb Points = segment Curb Points;
  End
  If Cand Curb Points= empty
    Prev. Off-Road = off-Road Point Cloud; Prev. Ang = road Ang;
    Continue;
  End
  Curb Points = Cand Curb Points. Location;
  If there is tracking
    Cand Ang = atan2d (curb Points (:, 2), curb Points(:,1));
    Cand Ang (Cand Ang < 0) = Cand Ang (Cand Ang < 0) + 360;
%Dividing into right and left curb points for the candidate curb points
    If (road Ang (1) >= 270 || road Ang (1) < 90) && (road Ang (2) >= 90 && road Ang (2) < 270)
      idx = Cand Ang >= road Ang (1) & Cand Ang < road Ang(2);
    Else
      Idx = Cand Ang >= road Ang (2) | Cand Ang < road Ang (1);
    End
    Right Cand Curb Points = curb Points (idx, :);
    Left Cand Curb Points = curb Points (idx, :);
%Computing right/left curbs to polynomials for tracking and visualizing
    Right Poly = fit Poly RANSAC (right Cand Curb Points (: 1:2), 2, 0.1);
    Left Poly = fit Poly RANSAC (left Cand Curb Points (: 1:2), 2, 0.1);      Point Cloud = select (point
Cloud, ROI Idx);
  Else
    Select (point Cloud, ROI Idx) & Output Point Cloud;
  End
  Cand Curb Queue = [Cand Curb Queue; Cand Curb Points];
  If Cand Curb Queue Count <= Cand Curb Queue Size
    Cand Curb Queue Count = Cand Curb Queue Count + 1;
  Else
    Cand Curb Queue = Curb Queue (2: end);
  End
  Prev. Off-Road Point Cloud = off-Road Point Cloud;
  If player= Open
    View (player, output Point Cloud);
  End
End
```

Algorithm 1 (Tracking Curb Points algorithm)Algorithm 1 Curb points tracking

2.6 Examine the smoothness and drift of the curb tracking

We plot the curb path recognition to produce a data image and compare it to the polynomial curbs that were tracked. Each plot contrasts the parameters using the Kalman filter and without it. Comparing the curbs' drift along the y-axis in the first picture, and the curb polynomials' smoothness is compared in the second. The rate of variation in the slope of the curb polynomial is known as smoothness. According to [21], the feature of a tangent angle is distinct as that angle produced by two vectors and defined by:

$$\theta_i = cos^{-1}\left(\frac{V_i^R V_i^L}{|V_i^R||V_i^L|}\right) \tag{2}$$

3 Results and discussion

The ROI from the point clouds is classified and categorized by the points inside it, which includes off-road and on-road points as a pre-handing step for finding the curb line. The visualization for the ground truth labels of the point cloud, after selecting the dataset 1st frame for more processing, and the point cloud over off-road and on-road points are shown in Fig. 6.

The obtained smoothness feature examines the area's smoothness close to a point. The lower value means that the point is a plane pinpoint, and a higher smoothness value means an edge point. The height dissimilarity characteristic examines the maximum difference and the standard deviation of the altitude in the region of that point, while the Vertical and Horizontal Continuity Attribute characteristics inspect the continuity of a point according to its nearest adjacent point. The feature curb points are the points that meet the attribute criteria for the aforementioned characteristics. From the mark curb points, the applicant curb points are calculated.

The tracking curb point's stage includes looping via the Lidar and processing data to track and extract the prospect curb points. Curb tracking can be stopped at segmented roadways and resumed when the ego vehicle travels off of those roads. Figure 7 demonstrates the detection of candidate curb points and the tracking of the curb point's process.

In order to examine the smoothness and drift of the curb tracking, curb path recognition to produce data image and compare it to the polynomial curbs that were tracked is shown. Figure 8 shows the evaluation of the curbs' drift over the y-axis, and the curb polynomials' smoothness. The rate of variation in the slope of the curb polynomial is known as smoothness.

The results of Fig. 8 demonstrate that the drift values were approximately constant with the filtered drift values along the Y-axis (m). This difference increases between (25–30) m far from the vehicle from both sides the left and right of the car. The maximum drift in curb points was 1.62 m for the left curb drift, while it was 0.87 m for the right curb drift, which is very close to that obtained in [11].

The bottom results of Fig. 7 demonstrate that the curve smoothness for the left side has a good agreement with the filtered one. However, the maximum difference with about 10 m at a distance of 25 m with respect to the vehicle. In contrast, the curve smoothness values for the right side have a better agreement than the left side one with the corresponding filtered one. However, the maximum difference was about 3.72 m only at the distance of 27 m with respect to the vehicle.

4 Method evaluation

With referring to the ground truth of the considered dataset, we calculated the Accuracy, Precision, and Recall metrics for evaluating performance. The formulas can be defined as follows:

- 1. The accuracy is defined as the overall correctness of the model, which is computed by the ratio of correctly classified curb points:

$$Accuracy = \frac{(True\ Positives + True\ Negatives)}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives)}$$

- 2. The precision reflects how many of the curb points identified by the model actually curbs are and can be given by:

$$Pricision = \frac{True\ Positives}{(True\ Positives + False\ Positives)}$$

- 3. The recall indicates how many of the actual curb points were correctly detected by the model and can be given by:

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)}$$

where in Lidar point cloud data of curb detection, True Positive (TP) represents a point that is correctly classified as a curb point, False Positive (FP) represents a point that

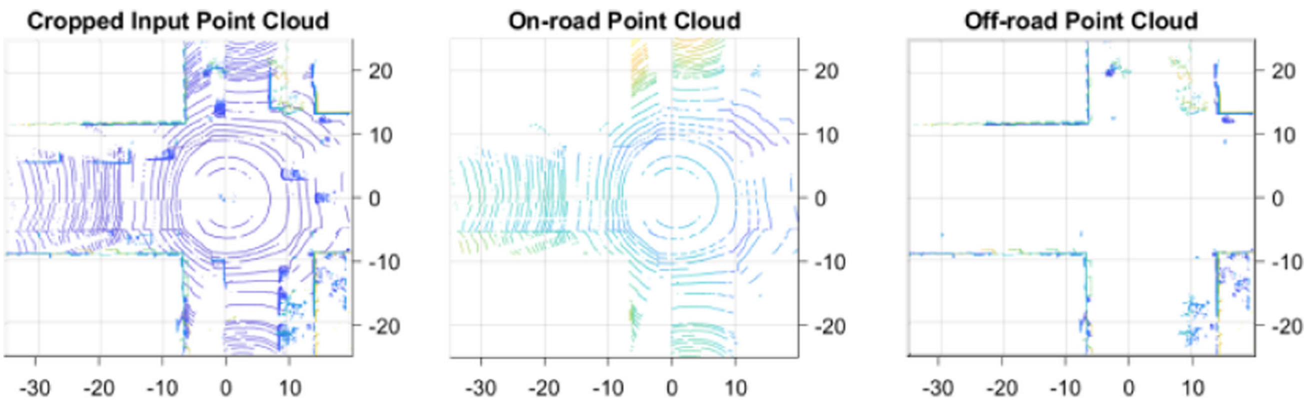


Fig. 6 Visualization for the point cloud over a cropped input point cloud, off-road point cloud, and on-road point cloud

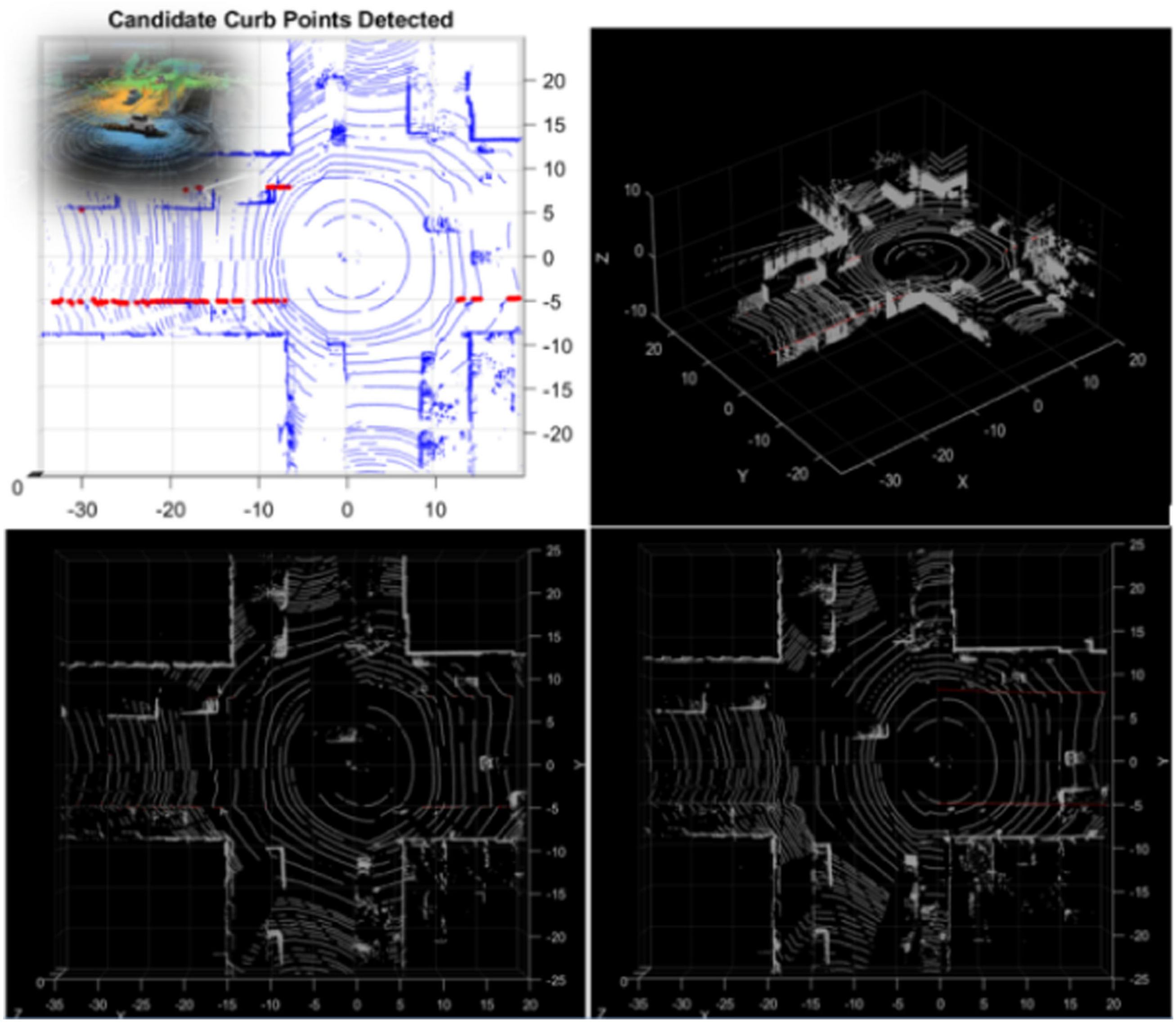


Fig. 7 Tracking process of curb points

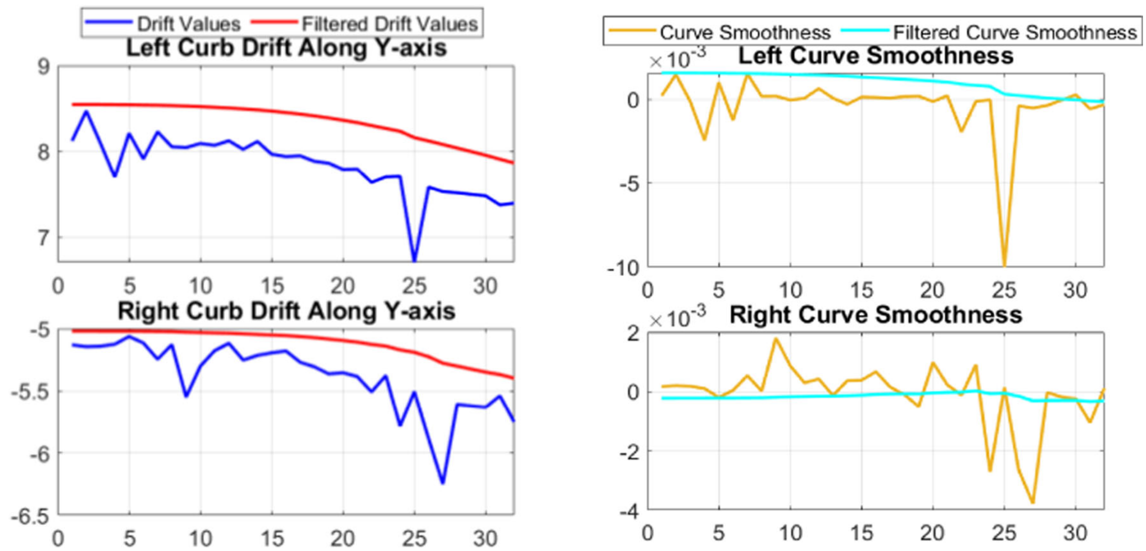


Fig. 8 The results of the evaluation of curbs' drift by comparing the filtered drift measurements with the drift values are on top, while the evaluation of the curb polynomial smoothness, which is a rate of variation in the slope of the curb polynomial, by comparing the curve smoothness and the filtered curve smoothness is on the bottom

is incorrectly classified as a curb point (an object near the curb or a noise point), True Negative (TN) represents a point that is correctly classified as a non-curb point (usually ground), and False Negative (FN) represents an actual curb point that is missed by the model (not classified as a curb point).

5 Conclusions

This work presents a real-time development method of curb point recognition and tracking using 3D point cloud data of Lidar sensors. A Kalman filter is employed to track these curbs while taking the motion of the vehicle into account.

Feature	Proposed Method (Lidar ROI)	Existing Techniques
Method	Extracts ROI, classifies points (ground vs non-ground), analyzes features (elevation, slope, normal) for curb detection	Manual feature extraction ([11, 19], and [22]) < br >—Deep learning based methods (3D U-Net[23], Cylinder 3D [24])
Strengths	Potentially efficient due to focusing on a specific region (ROI). May be robust to variations in ground surface (e.g., uneven terrain)	Manual methods offer interpretability but may struggle with complex data.—Deep learning methods can be data-hungry and require careful training
Weaknesses	Reliant on accurate ground segmentation and feature selection. Performance might be affected by complex curb shapes or objects near curbs	Manual methods might require domain knowledge for feature selection
Evaluation Metrics	Precision: 0.8782 Recall: 0.8695	Ref. / Precision / Recall [19] 0.7209 0.7013 [22] 0.6878 0.6864 [11] 0.6854 0.6564 [23] 0.7695 0.7492 [24] 0.8049 0.8038

The robustness and correctness of the suggested method are demonstrated by employing the PandaSet dataset. The real-time simulation demonstrates that the suggested method for curb tracking and identification is time-efficient and accurate. The curb drift results demonstrated that the drift values were approximately constant with the filtered drift values along the Y-axis (m) and the drift difference increases between (25–30) m far from the vehicle from both sides the left and right of the car. The curve smoothness results demonstrated that the curve smoothness for the left and right sides have a good agreement with the filtered one and the values for the right side have a better agreement than the left side one with the corresponding filtered one. The difficulty of finding a junction without curbs is still a challenge. To improve the detecting method's accuracy and reliability, additional changes such as a high-accuracy digital map should be correctly included.

As future work, some methods to integrate IoT with curb detection includes; (1) a role in data collection, where IoT devices could be utilized to collect additional data about the nearby location, such as radar data from sensors or images from cameras. This information might then be employed to advance the curb detection accuracy. (2) A role in real-time monitoring, which could be employed to provide warnings to drivers about upcoming hazards by renewing maps.

Declarations

Conflict of interest The authors have no conflicts of interest to declare.

References

1. Hütt C, Bolten A, Hüging H, Bareth G (2023) UAV LiDAR metrics for monitoring crop height, biomass and nitrogen uptake: a case study on a winter wheat field trial. PFG J Photogramm Remote Sens Geoinform Sci. https://doi.org/10.1007/s41064-022-00228-6
2. Zhang Y, Zou S, Liu X, Huang X, Wan Y, Yao Y (2022) LiDAR-guided stereo matching with a spatial consistency constraint. ISPRS J Photogramm Remote Sens. https://doi.org/10.1016/j.isprsjprs.2021.11.003
3. Santos MF, Victorino AC and Pousseur H (2023) Model-based and machine learning-based high-level controller for autonomous vehicle navigation: lane centering and obstacles avoidance. IAES Int J Robot Autom. https://doi.org/10.11591/ijra.v12i1.pp84-97
4. Suleymanov T, Kunze L and Newman P (2019) Online inference and detection of curbs in partially occluded scenes with sparse LIDAR. In: 2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019. https://doi.org/10.1109/ITSC.2019.8917086
5. Horváth E, Pozna C, Unger M (2022) Real-time lidar-based urban road and sidewalk detection for autonomous vehicles. Sensors. https://doi.org/10.3390/s22010194
6. Bashi OID, Hameed HK, Al Kubaiaisi YM and Sabry AH (2023) Development of object detection from point clouds of a 3d dataset by point-pillars neural network. Eastern-Eur J Enterprise Technol. https://doi.org/10.15587/1729-4061.2023.275155
7. Shallal AH, Salman SA, Sabry AH (2022) Hall sensor-based speed control of a 3-phase permanent-magnet synchronous motor using a field-oriented algorithm. Indones J Electr Eng Comput Sci 27(3):1366–1374. https://doi.org/10.11591/ijeecs.v27.i3.pp1366-1374
8. Sun PP, Zhao XM, Xu ZG, Min HG (2018) Urban curb robust detection algorithm based on 3D-LIDAR. Zhejiang Daxue Xuebao (Gongxue Ban)/J Zhejiang Univ (Eng Sci). https://doi.org/10.3785/j.issn.1008-973X.2018.03.012
9. Panev S, Vicente F, De La Torre F, Prinet V (2019) Road curb detection and localization with monocular forward-view vehicle camera. IEEE Trans Intell Transp Syst. https://doi.org/10.1109/TITS.2018.2878652
10. 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 J Photogramm Remote Sens. https://doi.org/10.1016/j.isprsjprs.2017.06.007
11. Zhang Y, Wang J, Wang X, Dolan JM (2018) Road-segmentation-based curb detection method for self-driving via a 3D-LiDAR sensor. IEEE Trans Intell Transp Syst. https://doi.org/10.1109/TITS.2018.2789462
12. Huang R, Chen J, Liu J, Liu L, Yu B, Wu Y (2017) A practical point cloud based road curb detection method for autonomous vehicle. Information (Switzerland). https://doi.org/10.3390/info8030093
13. Liu T, Wang Y, Niu X, Chang L, Zhang T, Liu J (2022) LiDAR odometry by deep learning-based feature points with two-step pose estimation. Remote Sens. https://doi.org/10.3390/rs14122764
14. Wang M, Liu R, Lu X, Ren H, Chen M, Yu J (2020) The use of mobile lidar data and Gaofen-2 image to classify roadside trees. Meas Sci Technol. https://doi.org/10.1088/1361-6501/aba322
15. Kim DH, Kim GW (2021) Automatic multiple LiDAR calibration based on the plane features of structured environments. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3087266
16. Lv X, Wang S, Ye D (2021) CFNet: LiDAR-camera registration using calibration flow network. Sensors. https://doi.org/10.3390/s21238112
17. Xiao P, Shao Z, Hao S, Zhang Z, Chai X, Jiao J, Yang D (2021) PandaSet: advanced sensor suite dataset for autonomous driving. In: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. https://doi.org/10.1109/ITSC48978.2021.9565009
18. Maxwell AE, Pourmohammadi P, Poyner JD (2020) Mapping the topographic features of mining-related valley fills using mask R-CNN deep learning and digital elevation data. Remote Sens. https://doi.org/10.3390/rs12030547
19. Wang G, Wu J, He R, Tian B (2021) Speed and accuracy tradeoff for LiDAR data based road boundary detection. IEEE/CAA J Autom Sinica. https://doi.org/10.1109/JAS.2020.1003414
20. Hu J, Razdan A, Femiani JC, Cui M, Wonka P (2007) Road network extraction and intersection detection from aerial images by tracking road footprints. In IEEE Trans Geosci Remote Sens. https://doi.org/10.1109/TGRS.2007.906107
21. Byun J, Sung J, Roh MC, Kim SH (2011) Autonomous driving through Curb detection and tracking. In: URAI 2011 - 2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence. https://doi.org/10.1109/URAI.2011.6145975
22. Sun P, Zhao X, Xu Z, Wang R, Min H (2019) A 3D LiDAR data-based dedicated road boundary detection algorithm for autonomous vehicles. IEEE Access. https://doi.org/10.1109/ACCESS.2019.2902170

23. Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O (2016) 3D U-net: Learning dense volumetric segmentation from sparse annotation. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-319-46723-8_49
24. Jiang F, Gao H, Qiu S, Zhang H, Wan R, Pu J (2023) Knowledge distillation from 3D to bird's-eye-view for LiDAR semantic segmentation. In Proceedings - IEEE International Conference on Multimedia and Expo. https://doi.org/10.1109/ICME55011.2023.00076

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.