Engineering 4 (2018) 479–490

Contents lists available at ScienceDirect

Engineering

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

Research
Robotics—Article

A Co-Point Mapping-Based Approach to Drivable Area Detection for Self-Driving Cars

Ziyi Liu [a,b], Siyu Yu [a,b], Nanning Zheng [a,b,*]

[a] Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China
[b] National Engineering Laboratory for Visual Information Processing and Applications, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 6 April 2017
Revised 13 November 2017
Accepted 28 December 2017
Available online 18 July 2018

Keywords:
Drivable area
Self-driving
Data fusion
Co-point mapping

ABSTRACT

The randomness and complexity of urban traffic scenes make it a difficult task for self-driving cars to detect drivable areas. Inspired by human driving behaviors, we propose a novel method of drivable area detection for self-driving cars based on fusing pixel information from a monocular camera with spatial information from a light detection and ranging (LIDAR) scanner. Similar to the bijection of collineation, a new concept called co-point mapping, which is a bijection that maps points from the LIDAR scanner to points on the edge of the image segmentation, is introduced in the proposed method. Our method positions candidate drivable areas through self-learning models based on the initial drivable areas that are obtained by fusing obstacle information with superpixels. In addition, a fusion of four features is applied in order to achieve a more robust performance. In particular, a feature called drivable degree (DD) is proposed to characterize the drivable degree of the LIDAR points. After the initial drivable area is characterized by the features obtained through self-learning, a Bayesian framework is utilized to calculate the final probability map of the drivable area. Our approach introduces no common hypothesis and requires no training steps; yet it yields a state-of-art performance when tested on the ROAD-KITTI benchmark. Experimental results demonstrate that the proposed method is a general and efficient approach for detecting drivable area.

1. Introduction

Road detection has long been considered to be a decisive component of self-driving cars, and attracts wide research attention. Thus far, remarkable progress has been achieved in road detection [1–3]. However, automatic driving decisions that are based on road detection alone may fail to deal with certain emergencies in which the detected road becomes undrivable due to suddenly turning vehicles or pedestrians. In fact, when driving a car, a human driver understands scenarios by classifying obstacles versus non-obstacles, rather than merely identifying the road. Thus, a human driver can choose to drive on flat areas that are not normally viewed as roads for safety reasons during an emergency. For self-driving cars, instead of the detection of road areas, the detection of such "flat areas" can provide a more comprehensive knowledge for the decision-making process, allowing self-driving cars to act more like human drivers.

Although most existing road-detection methods are already available for well-marked roads through sample training, the problem of detecting a road surface on weakly marked roads and lanes in urban and rural environments remains unsolved, due to the high variability of the scene layout, illumination, and weather conditions. Thus far, no reliable solution exists; therefore, a robust and efficient method is urgently needed.

In image segmentation, the boundaries of objects generally appear in the area of depth discontinuities. Therefore, image segmentation should be fused with depth discontinuities. In projection geometry, a homograph is an isomorphism of projective spaces; it is a bijection that maps lines to lines, in what is known as collineation. Here, we introduce a new concept that is similar to the bijection of a collineation: co-point mapping. Co-point mapping is a bijection that maps points from a laser sensor to the points on the edges of image segmentation. The projective space is constructed as a set of points of normal vectors over a given field. Some co-points are not homographs owing to changes in illumination, unevenness of the road, and shadows in two-dimensional (2D) images. In order to overcome these problems, we simply

* Corresponding author.
E-mail address: nnzheng@mail.xjtu.edu.cn (N. Zheng).

https://doi.org/10.1016/j.eng.2018.07.010

use the normal vector of the point cloud data instead of using the raw point cloud data, which will be described later. In this context, pixel-depth data fusion can be clearly described, and the homographs of pixel depth are defined as co-point mapping.

We thus propose a self-adaptive method for drivable area detection by leveraging co-point mapping to fuse the pixel information from a monocular camera with the spatial information from a laser sensor, as shown in Fig. 1. By combining the image pixels' coordinates with the spatial location of each laser point, a Delaunay triangulated graph [4] is built to establish the spatial relationship among the laser points, and the normal vectors of the triangles are used in the obstacle classification task of the laser points. Next, initial drivable areas are located by fusing obstacle classification results with image superpixels through self-learning. Candidate drivable areas in different feature spaces can then be obtained. These features are: the drivable degree (DD) feature, the normal vector (NV) feature, the color feature, and the strength feature. Finally, a Bayesian framework is utilized to fuse the candidate areas in order to obtain the final drivable area. In our evaluations, we tested our method using the ROAD-KITTI benchmark [5]. Our results, when compared with other fusing methods, demonstrated that the proposed method achieves state-of-the-art results without requiring training or assumptions about shape or height; this result validates our method as being robust and having a high generalization ability.

The key contributions of this paper are as follows:

- We propose an unsupervised detection method based on data fusion without the need for a strong hypothesis, which ensures our method's generalization ability in different urban traffic scenes.
- We introduce the new concept of co-point mapping, which describes a novel kind of constraint in the fusion of data from the laser sensor and camera.

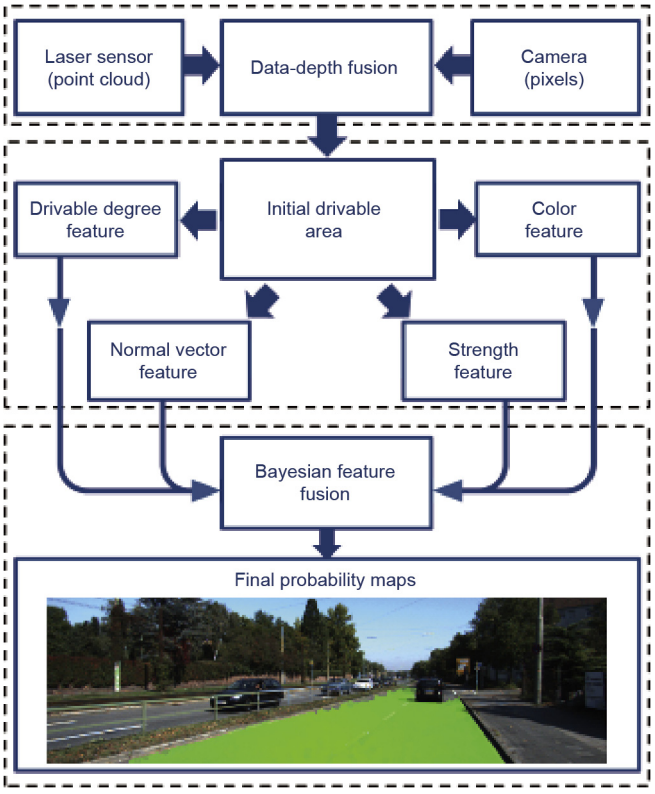

Fig. 1. Framework of the proposed detection method, which can be divided into three general steps: data fusion, feature extraction, and feature fusion.

- We design a new feature called DD in order to describe the drivable degree of the laser points.

2. Related work

A robust road area detection method is central to self-driving cars. Many methods have been proposed over the past decades to deal with this problem. These methods can be categorized by the sensors that are used to acquire data, which include monocular cameras, stereo vision, laser sensors, and the fusion of multi-sensors.

Monocular vision-based approaches have been widely used in road detection. Compared with other sensors, a visual sensor is small in size, low in cost, and easy to install. Moreover, rich visual information is available from a visual sensor, which has a wide range of detection. In addition, the concealment capability of a visual sensor is better than those of other sensors. Above all, the principle and structure of a visual sensor are similar to those of human sensory organization. In road detection, 2D information on the visual scene, such as color, corner points, texture, edge, and shape, are utilized. Regarding color cue, it is common to process segmentation in the RGB color space [6], HSI color space [7], or other color spaces. Jau et al. [8] compared RGB and HSI color segmentation under different lighting conditions. Finlayson et al. [9] presented a physics-based illumination invariant space that achieves a shadow-free image representation, which was used in this paper. Moreover, by exploiting the spectral properties of the camera that is used to capture raw color images, Maddern et al. [10] proposed another illumination invariant color space that reduces the effects of illumination variation caused by sunlight.

Another hot research topic is convolutional neural network (CNN)-based methods, which have achieved great success in this field [11,12]. Originally, CNNs were used to solve classification problems [13,14]; however, with the emergence of recent work [15,16], the utilization of CNN-based methods for sematic segmentation has entered a period of upsurge.

However, the conception of a road is different from other vision conceptions because the pixel appearance in vision is not the only criterion to detect a road area. Physical attributes, such as flatness, contribute more to the conception of a road, which indicates that methods that rely only on monocular vision are not reliable enough for self-driving cars. Although CNN-based methods can achieve good performance, they heavily depend on training, may fail to deal with unseen scenarios, and can have overfitting problems. Unlike problems such as scene categorization [17] or similarity learning [18], road area detection is an ill-conditioned problem that requires using 2D information to solve a task in a three-dimensional (3D) real-world scene. Although many 3D cues, such as the horizon line and vanishing point, are used to alleviate this problem [19–21], the detection of these 3D cues is in itself an unsolved problem [22,23]; some geometric assumptions may result in failure or may reduce the generalization ability of the methods, as shown in Fig. 2.

In recent years, the advent of sensors has inspired the development of many road-detection methods based on laser sensors that can offer supplementary depth measurement of real-world 3D scenes. These methods, which use the spatial locations of laser points to analyze a scene and identify flat areas as road, can be classified as follows:

(1) **Grid-based methods.** Since point clouds contains a large quantity of data, 2D grid-based methods [24–26] are commonly used to reduce the data size, while statistics of the points within a grid are calculated in order to characterize each grid, and may include the average height and the maximum height difference. Although these methods are straightforward, noise-robust, and efficient, the selection of appropriate thresholds is difficult.

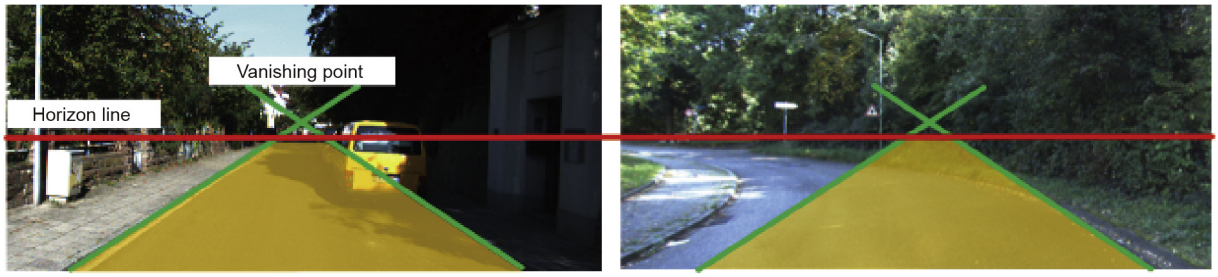

Fig. 2. A comparison of different road scenes in which 3D cues may work or fail. (a) A situation in which 3D cues work well; (b) a situation in which 3D cues may fail because of the shift of the vanishing point and the curved shape of the road.

(2) **Plane-fitting-based methods.** The basic assumption in these methods is that the road is flat and smooth, so that it can be fitted by a plane with several parameters [27,28]. Typical plane-estimation methods are well developed, such as random sample consensus (RANSAC) [29]. However, these methods maybe suffer in heavy traffic scenarios because of the lack of true ground laser points.

(3) **Methods based on the spatial relationship between neighboring points.** These methods [30,31] take advantage of the spatial relationship between neighboring points to extract features (such as the normal vector) or probabilistic models in order to estimate the ground laser points or obstacle laser points (such as those of the curb).

All methods that are based on laser sensors suffer from the sparsity of point cloud data; as a result, it is difficult to reconstruct details from the laser points.

Detecting the road area can be regarded as a two-class labeling problem, and the conditional random field (CRF) framework is popular in this area [32,33]. The CRF framework formulates a labeling problem as a calculation of the maximum posterior probability of the overall labeling result, given the observations from all aspects. This is a general framework in which different observations can be defined by designing different entries of the energy function and potential function. Thus, CRF-based methods are widely used in fusion methods in order to both balance the data from different sources and obtain optimal fusion results [34,35]. However, the computation and memory consumption of CRF-based methods are large, and manually labeled data are required.

To overcome the drawbacks mentioned above, this paper proposes a co-point mapping-based self-adaptive method of drivable area detection by fusing data from a laser sensor with data from a monocular camera. First, our method conducts several preprocessing steps, which include: conducting superpixel segmentation to obtain the minimum image-processing units for subsequent steps; projecting the laser points onto an RGB image via cross-calibration and co-point mapping; and utilizing the Delaunay triangulation properties to preprocess the laser points in order to establish a spatial relationship. Next, pixel-depth data fusion is processed by leveraging the preprocessing results, and obstacle classification can be performed by utilizing the data fusion results. By combining superpixels with the obstacle classification results, an initial drivable area is located. This step is followed by feature-extraction processing, by which we obtain the DD feature, NV feature, color feature, and strength feature. All these features can be easily transformed into probabilities in a self-learning manner. Finally, a joint probability can be calculated superpixel by superpixel, by leveraging the Bayesian framework to obtain a joint probability map of the drivable areas.

Our proposed method distinguishes itself from other detection techniques in three main aspects:

- Our method requires no strong hypothesis, training processing, or labeled data.
- By leveraging the co-point perspective and fusing data from both a laser sensor and a monocular camera, our method is robust to variation in illumination and can cope with complex scenarios.
- Our method adopts superpixel segmentation, and superpixels are then taken as the minimum processing elements. The advantages of replacing pixels with superpixels will be detailed later.

Unlike methods that lack feature-level fusion [36,37], our proposed method combines monocular vision with laser sensors to obtain abundant information on both the data level and the feature level. Our method extracts and fuses features in a self-learning way. In addition, co-point perspective and superpixel representation are utilized to make the method more robust and efficient. As demonstrated by the experimental results, our method achieves higher accuracy than other methods. Thus, we consider our method to be a general, practical, and self-adaptive approach to the detection of drivable areas for self-driving cars.

3. Preliminary knowledge

In this section, we provide the preliminary knowledge that is required by our method, including superpixel representation of images, the projection of laser points onto an RGB image, and the establishment of log-chromaticity space.

3.1. Superpixel representation

The idea of superpixels was originally developed by Ren and Malik [38]; a superpixel is a group of pixels that is coherent in color or texture, such that a superpixel representation preserves most of the structural information of the original image.

In this paper, superpixels, rather than pixels, are taken as the minimum processing units in image-processing steps, and assist in shaping candidate areas. Because of the performance improvement of superpixel methods, replacing pixels with superpixels reduces the computation and memory costs without sacrificing much accuracy. In addition, the usage of superpixels takes color information into account and achieves robust results when dealing with situations with complex illumination.

In order to segment the original images better, the superpixel method should meet two requirements: First, the speed of superpixel generation should be fast; and second, the generated superpixels should "stick" to the edges.

As proposed in Refs. [39,40], we utilized sticky-edge adhesive superpixels, which meet our requirements. This method is an improved version of the simple line interface calculation (SLIC)

method [41], by the addition of an edge term. With this added edge term, the generated superpixels better adhere to the edges, thus preserving more image structure and resulting in better object boundaries.

3.2. Laser points project processing

As shown in Fig. 3, laser sensor coordinates are intended to be projected into the camera coordinates, as presented in Ref. [42]. The projection of a 3D point $p_{\_laser} = (x_l, y_l, z_l, 1)^T$ in the laser sensor coordinate to a point $p_{\_camera} = (x_c, y_c, z_c, 1)^T$ in the camera coordinate is given as:

$$p_{\_camera} = \mathbf{R}^0_{rect}\mathbf{T}^{cam}_{velo}p_{\_laser} \tag{1}$$

where $\mathbf{R}^0_{rect}$ is the rotation matrix. Here, $\mathbf{R}^0_{rect}$ is expanded into a 4×4 matrix by appending a fourth zero row and column, and by setting $\mathbf{R}^0_{rect}(4,4) = 1$. $\mathbf{T}^{cam}_{velo}$ is the transformation matrix, and is obtained by:

$$\mathbf{T}^{cam}_{velo} = \begin{pmatrix} \mathbf{R}^{cam}_{velo} & \mathbf{t}^{cam}_{velo} \\ 0 & 1 \end{pmatrix} \tag{2}$$

where $\mathbf{R}^{cam}_{velo}$ and $\mathbf{t}^{cam}_{velo}$ are the rotation matrix and the translation vector, respectively, as given in Ref. [42]:

$$(u, v, 1)^T = \mathbf{P}^{(i)}_{rect}(x_c, y_c, z_c, 1)^T \tag{3}$$

where $\mathbf{P}^{(i)}_{rect}$ is the projection matrix from the ith image plane (we use a second camera in our work) for all points p_c, from which the location information of the camera coordinates, (u_i, v_i), is obtained.

After the projection and rectification, we obtained a set of laser points, $\mathbb{P} = \{\mathbf{P}_i\}_{i=1}^N$, where $\mathbf{P}_i = (x_i, y_i, z_i, u_i, v_i)$.

3.3. Log-chromaticity color space

As presented in Ref. [36], in order to obtain color features that are independent of shadows and lighting conditions, we transformed RGB color images (I) into log-chromaticity spaces (I_{log}), in order to generate an illuminant-invariant image I_{l-c}. Each $\{\log(R/G), \log(B/G)\}$ pixel value in the log-chromaticity space corresponds to a $\{R, G, B\}$ pixel value in the original RGB image. As shown in Fig. 4, we obtain a grayscale image I_{l-c} by projecting the $\{\log(R/G), \log(B/G)\}$ pixel values along an orthogonal axis defined with the angle θ. The angle θ is defined as the invariant direction orthogonal to the lighting change lines; it is device dependent and can be calibrated. We set θ to 45°, as suggested in Ref. [43]. I_{log} can be calculated as follows:

$$I_{log}(u, v) = \frac{\log[I_R(u, v)/I_G(u, v)] + \log[I_B(u, v)/I_G(u, v)] \cdot \tan\theta}{1 + \tan^2\theta} \tag{4}$$

4. Pixel-depth data fusion

Rather than using sensors that can directly detect obstacles [44], we try to find obstacles by leveraging the fusion of the spatial location information provided by a laser sensor with the appearance information provided by a monocular camera. This section presents a series of processes that are designed for pixel-depth data fusion. The first process fuses superpixels with laser points for efficiency and robustness. The second process, co-point mapping, involves fusing points from the laser sensor with points on the edge of the image segmentation, in order to eliminate laser points within the flat area. The third and final process involves the fusion between the spatial information and the image coordinates; this step models the spatial relationship of the laser points and the obstacles by generating an undirected graph.

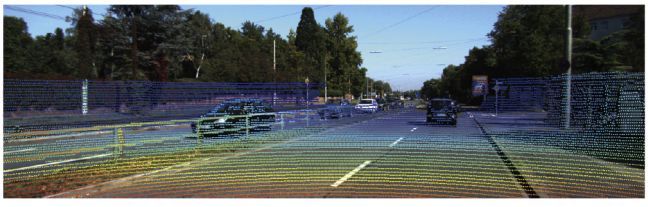

Fig. 3. An example of laser points and image pixels alignment. The color of each laser point represents its depth, so the image is better viewed in color.

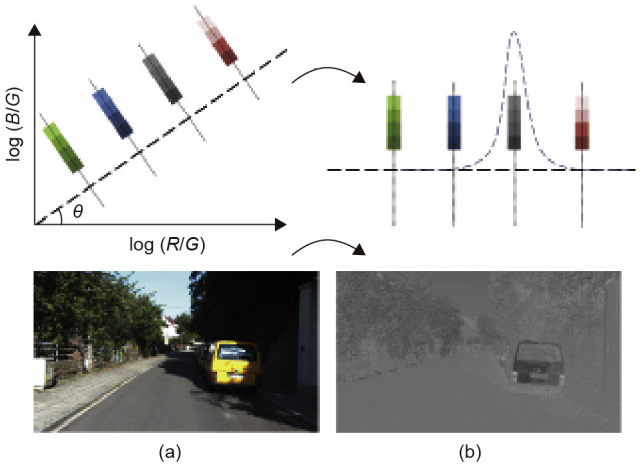

Fig. 4. Examples of color space transformation from (a) RGB space to (b) log-chromaticity space. It can be seen that the shadow is successfully removed but the color difference between the road and other objects (i.e., the yellow car and green plants) is well-preserved.

4.1. Image processing with superpixel representation

Our method adopts the superpixel representation described in Section 3.1. In addition to the advantages mentioned above, it was found that superpixels and laser points are complementary. First, laser points contain spatial location information that cannot be obtained from a monocular camera; second, superpixels are dense, so that robust local statistics of both pixels and laser points can be calculated. Superpixels also contain color information that cannot be captured by a laser sensor. Therefore, instead of pixels, superpixels were taken as the minimum units in the image-processing steps. In Section 5, superpixel segmentation will be used to learn features of the initial drivable area.

4.2. Bijection of edge information via co-point mapping

Since the laser points and image pixels are both provided through observation of the same scene at the same time, they are reflections of the same structure. Therefore, the projection between them should satisfy certain constraints. Inspired by the conception of collineation in projection geometry, we introduce a new concept, co-point mapping, to describe this kind of constraint. Similar to the bijection of collineation, co-point mapping represents a bijection that maps points from a laser sensor to points on the edge of the image segmentation.

Using co-point mapping, we can improve the alignment performance and eliminate undesired laser points. More specifically, edges can be regarded as the basic element that affects both the appearance of pixels and the structure of laser points. Bijection of edge information between pixels and laser points is the key in data fusion. When aligning pixels and laser points, edges must be

aligned correctly. Using cross-calibration, an initial alignment can be obtained. Next, we can improve that alignment through co-point mapping. Similarly, bijection of edge information can eliminate redundant laser points that are located in a flat area. It can accelerate the whole process and improve the robustness of our algorithm, since there may be noisy laser points within the flat area.

In this paper, we only use co-point mapping to eliminate laser points within the flat area, as shown in Fig. 5. First, we use the edges of all superpixels to form an edge pool of the whole image. Obviously, there are many redundant edges. However, this is a reasonable practice in order to retain all the true edges in the pool. In practice, image dilation is used to overcome alignment errors. Next, laser points located in the edge pool are kept for further processing, while others are discarded. The experimental results showed that 27% of the laser points were eliminated without sacrificing too much accuracy; this finding demonstrates that co-point mapping is feasible and efficient.

4.3. Obstacle classification

The obstacle classification step can be formulated as finding the mapping function, $ob(\mathbf{P}_i)$, where:

$$ob(\mathbf{P}_i) = \begin{cases} 1, & \text{if } \mathbf{P}_i \text{ is an obstacle point} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The classification result is shown in Fig. 6.

We assume that the value of $ob(\mathbf{P}_i)$ only depends on the flatness of the surface around $\mathbf{P}_i$, which in the real world is reflected by the laser points. Thus, the obstacle classification problem is decomposed into two sub-problems: how to find the surface around $\mathbf{P}_i$ and how to determine whether it is flat or not.

The first sub-problem is solved by leveraging a Delaunay triangulation [4]. A Delaunay triangulation for a set of points in a plane is a triangulation for which no point in the set is inside the circumcircle of any triangle in the triangulation. Its properties are such that each point has six surrounding triangles, on average. The nearest neighbor graph is a subgraph of the Delaunay triangulation, so it can be used to build a spatial relationship among the laser points. For each $\mathbf{P}_i = (x_i, y_i, z_i, u_i, v_i)$, we use its image coordinate frame (u_i, v_i) in a planar Delaunay triangulation to build an undirected graph $G = \{\mathbb{P}, E\}$, where E is the set of edges representing the spatial relationships among $\mathbf{P}_i$. Then, the surface around $\mathbf{P}_i$ is composed of surfaces (triangles) determined by $\{(u_j, v_j) | j = i \text{ or } \mathbf{P}_j \in Nb(\mathbf{P}_i)\}$, where $Nb(\mathbf{P}_i)$ is the set of points connected to $\mathbf{P}_i$. The edge $(\mathbf{P}_i, \mathbf{P}_j)$ is eliminated if it does not satisfy the following:

$$\|(\mathbf{P}_i - \mathbf{P}_j)\| < \varepsilon \tag{6}$$

where $\|(\mathbf{P}_i - \mathbf{P}_j)\|$ is the Euclidean distance of (x_i, y_i, z_i) and (x_j, y_j, z_j), and ε is the maximum length of any edge.

For the second sub-problem, the flatness of the surface around $\mathbf{P}_i$ can be measured by calculating the normal vector of its neighbor triangles. Next, the normal vector $N(\mathbf{P}_i) = (x_i^n, y_i^n, z_i^n)$ is calculated by averaging the normal vectors of these triangles. Then $ob(\mathbf{P}_i)$ is obtained by the following:

$$ob(\mathbf{P}_i) = \begin{cases} 1, & \text{if } \arcsin(z_i^n / \|Nb(\mathbf{P}_i)\|) > c \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where c is a manually set parameter, $Nb(\mathbf{P}_i)$ is the set of points connected to $\mathbf{P}_i$, and the normal vector of $\mathbf{P}_i$ is $N(\mathbf{P}_i) = (x_i^n, y_i^n, z_i^n)$. Eq. (7) can be explained as follows: If the angle between $Nb(\mathbf{P}_i)$ and the horizontal plane is larger than c, $\mathbf{P}_i$ will be classified as an obstacle point; otherwise, it will not be classified so. Thus, c denotes the maximum angle threshold, which is set to 60° in the experiment. The whole process is visualized in Fig. 7.

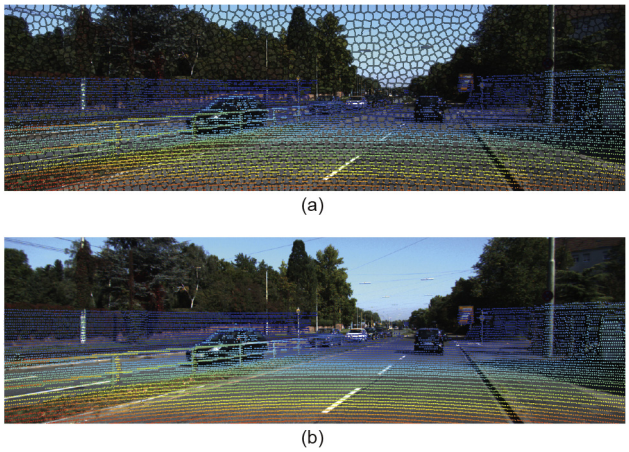

Fig. 5. An example of co-point mapping. Unlike (b), the image in (a) keeps only the laser points that are near the edges of superpixels. Although many points are discarded, the structure is successfully preserved.

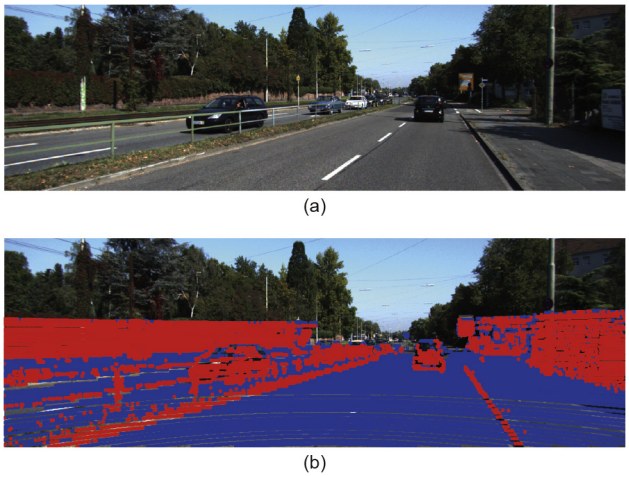

Fig. 6. The result of obstacle classification. This result shows that our method can detect obstacles (i.e., curbs, walls, and cars) well and is not affected by pixel appearances.

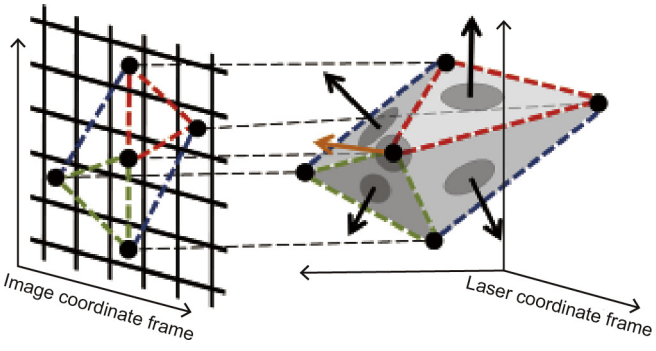

Fig. 7. A visualization of the obstacle classification step. Each black dot stands for a laser point. The relationships between the points are derived by planar Delaunay triangulation. The triangles' normal vectors are then calculated in the laser sensor coordinate frame. (Black arrows: the normal vectors of the neighbor triangles; orange arrow: the expected normal vector of the central black dot, which is calculated by averaging the surrounding black arrows)

5. Detecting drivable areas

This section explains the key step of our proposed method. First, a direction ray map (I_{DRM}) is obtained from the obstacle classification result. Next, an initial drivable area is obtained by combining I_{DRM} with superpixels. Afterward, the initial drivable area is characterized by features from different observations. The probabilities of each superpixel within the initial drivable area being foreground are then computed in a self-learning way. Finally, a Bayesian framework is utilized to derive the probability map of drivable areas.

5.1. Generating an initial drivable area

5.1.1. Detecting ray

The direction ray map I_{DRM} is generated as shown in **Algorithm 1**.

Algorithm 1. Generating direction ray map

Input:

The set of laser points $\{\mathbb{P}^{(h)}\}_{h=1}^{H}$;

Output:

Direction ray map, I_{DRM}

Initial I_{DRM} with the size of I and zeros elements
for $h = 1$ to H **do**
  Find obstacle set $\mathbb{O} = \{\mathbf{P}_i \,|\, \mathbb{o}\mathbb{b}(\mathbf{P}_i) = 1 \ \&\ \mathbf{P}_i \in \mathbb{P}^{(h)}\}$
  if $\mathbb{O} = \varnothing$ **then**
    $\mathbf{P}_{\mathrm{bin}}^{(h)} = \arg\max_{\mathbb{P}^{(h)}} dist(\mathbf{P}_i^{(h)}, \mathbf{P}_{\mathrm{base}})$
  else
    $\mathbf{P}_{\mathrm{bin}}^{(h)} = \arg\min_{\mathbf{P}_i^{(h)} \in \mathbb{O}} dist(\mathbf{P}_i^{(h)}, \mathbf{P}_{\mathrm{base}})$
  end if
  Line point $\mathbf{P}_{\mathrm{bin}}^{(h)}$ with point $\mathbf{P}_{\mathrm{base}}$ in I_{DRM}
end for

First, every $\mathbf{P}_i$ is transformed to polar coordinates in order to better represent the drivable area. That is, (u_i, v_i) is transformed into the polar coordinate whose origin point is the middle bottom pixel of the image (noted as $\mathbf{P}_{\mathrm{base}}$). Thus, $\mathbb{P}$ is represented as $\{\mathbb{P}^{(h)}\}_{h=1}^{H}$, where $\mathbb{P}^{(h)} = \{\mathbf{P}_i^{(h)}\}_{i=1}^{N^{(h)}}$ and $\mathbf{P}_i^{(h)}$ represents a point belongs to the hth angle range. Because of the sparsity of $\mathbb{P}$, it is necessary to address two problems: first, how to get over the "leakage" problem, as shown in Fig. 8; and second, how to obtain dense pixel areas from sparse rays.

For the former problem, the solution is provided by filtering the rays' length calculated in the image coordinate frame, as shown in Fig. 8(d). Since the width of a car is not negligible, the question of whether a region that is represented by one ray is drivable or not depends on how wide that region is. In other words, if a region is too narrow for a car to pass, no matter how flat that region may be, that region cannot be regarded as drivable. Thus, minimum filtering is adopted and the "leakage" problem is solved, as shown in Fig. 8(c).

For the latter problem, the most straightforward solution is to increase the number of H; however, doing so will aggravate the former problem. Therefore, we combine I_{DRM} with superpixels to obtain the initial drivable area. As mentioned in Section 4.1, this solution has two advantages: First, it greatly reduces the amount of data by replacing pixels with superpixels; and second, it fuses depth and color information.

After the combination of I_{DRM} with superpixels, the initial drivable area is represented by a set of superpixels defined as $\mathbb{S}_{\mathrm{int}} = \{S_i | S_i \cup I_{\mathrm{DRM}} \neq \varnothing\}$, and $\mathbb{P}_{S_i}$ represents the set of laser points located in S_i. Therefore, all of the following features can be computed based on superpixels instead of pixels, thereby introducing robust local statistics and accelerating the entire algorithm.

5.1.2. Generating the DD feature

To detect drivable areas, features that can describe the drivable degree must be defined and well designed. In this paper, the DD feature is proposed, as shown in **Algorithm 2**.

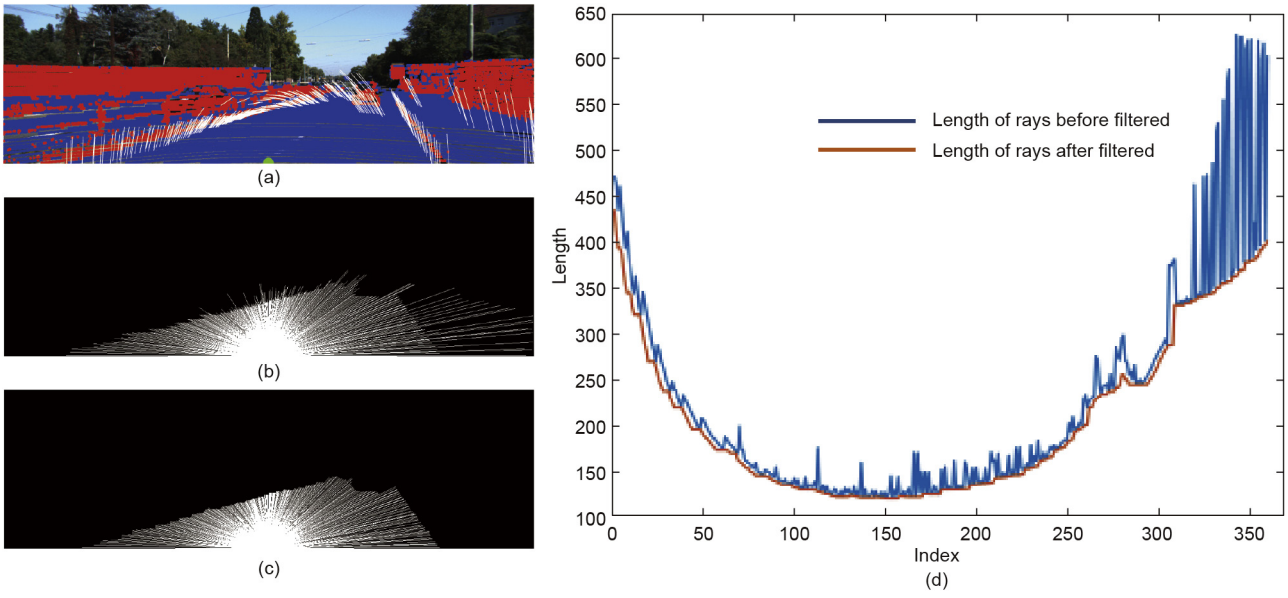

Fig. 8. An example of a "leakage" problem. (a) The "leakage" problem: the green dot is $\mathbf{P}_{\mathrm{base}}$; (b) In this image, the detecting rays are represented by white lines; (c) the result of minimum filtering; (d) a comparison of each ray's length before (blue curve) and after (orange) being processed by minimum filtering.

Algorithm 2. Getting DD feature

Input:

The set of laser points $\{\mathbb{P}^{(h)}\}_{h=1}^{H}$;

Output:

DD feature for every point $D(\mathbf{P}_i^{(h)})$

Initial $D(\mathbf{P}_i^{(h)})$ with the zero $(h = 1 \rightarrow H, i = 1 \rightarrow N^{(h)})$
for $h = 1$ to H **do**
  for $i = 1$ to $N^{(h)}$ **do**
    if $ob(\mathbf{P}_i^{(h)}) = 1$ **then**
      for $j = 1$ to $N^{(h)}$ **do**
        $D(\mathbf{P}_j^{(h)}) = D(\mathbf{P}_j^{(h)}) + abs(z_{\mathbf{P}_i^{(h)}} - z_{\mathbf{P}_{i-1}^{(h)}})$
      end for
    end if
  end for
end for

Points in $\mathbb{P}^{(h)}$ are arranged by their distances to $\mathbf{P}_{\text{base}}$ in the image coordinate frame, which signifies that all points in $\mathbb{P}^{(h)}$ satisfy the following:

$$dist(\mathbf{P}_i^{(h)}, \mathbf{P}_{\text{base}}) > dist(\mathbf{P}_{i-1}^{(h)}, \mathbf{P}_{\text{base}}),\ i = 2 \rightarrow N^{(h)} \tag{8}$$

Fig. 9 provides a schematic diagram of the DD value calculation. Fig. 10(c) visualizes what occurs when the DD feature is taken advantage of [45].

5.2. Obtaining self-adaptive feature models

Based on the obtained initial drivable area, candidate drivable areas are self-adaptively learned from four features in different probability spaces: the DD, NV, color, and strength features.

5.2.1. The DD feature

The DD feature of each superpixel ($D(S_i)$) is calculated as follows:

$$D(S_i) = \frac{1}{\|\mathbb{P}_{S_i}\|}\sum_{\mathbb{P}_{S_i}} D(\mathbf{P}_i) \tag{9}$$

Since $D(\mathbf{P}_i)$ is related to the difference in height, it is considered that the smaller $D(\mathbf{P}_i)$ is, the more drivable the corresponding area that $\mathbf{P}_i$ represents will be. The self-adaptive probability space of the DD feature used to locate the candidate drivable area is built through a Gaussian-like model as follows:

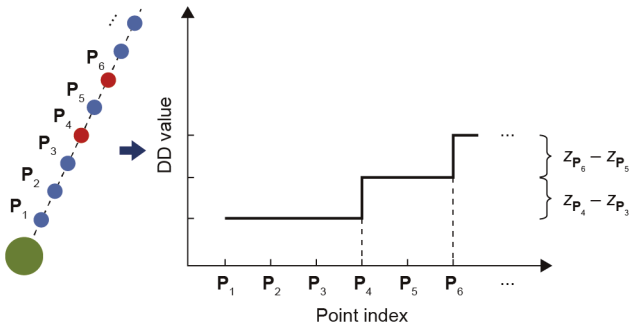

Fig. 9. A schematic diagram of DD calculation. The black dotted line on the left represents a ray in I_{DRM}. Blue dots represent non-obstacle laser points, red dots represent obstacle laser points, and the big green dot represents $\mathbf{P}_{\text{base}}$. In this figure, $\mathbf{P}_4^{(h)}$ is classified as an obstacle point; next, the difference of height between $\mathbf{P}_4^{(h)}$ and $\mathbf{P}_3^{(h)}$ is added to the DD value of all points after $\mathbf{P}_4^{(h)}$, so as to obtain $\mathbf{P}_6^{(h)}$.

$$D_{\text{prob}}(S_i) = \begin{cases} \exp\{-[D(S_i) - \mu_{\text{D}}]^2 / 2\sigma_{\text{D}}^2\}, & \text{if } D(S_i) \geq \mu_{\text{D}} \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where μ_{D} and σ_{D}^2 are the parameters of the Gaussian-like distribution that can be obtained by utilizing $\mathbb{S}_{\text{int}}$ without the need for manual setting or training. $D_{\text{prob}}(S_i)$ represents the probability of S_i belonging to the drivable area, given the DD feature observation.

5.2.2. The NV feature

According to Section 4.3, it is believed that the larger the value the normal vector has in z_i^n, the more drivable the surface is. Therefore, the NV feature of each superpixel, $N(S_i)$, is calculated using the minimum value of z_i^n Among $\mathbb{P}_{S_i}$. Similar to $D_{\text{prob}}(S_i)$, a self-adaptive probability feature space based on a Gaussian-like model with parameters μ_n and σ_n^2 is generated as follows:

$$N_{\text{prob}}(S_i) = \begin{cases} \exp\{-[N(S_i) - \mu_{\text{n}}]^2 / 2\sigma_{\text{n}}^2\}, & \text{if } N(S_i) \geq \mu_{\text{n}} \\ 1, & \text{otherwise} \end{cases} \tag{11}$$

where $N_{\text{prob}}(S_i)$ is the probability of S_i belonging to the drivable area, given the NV feature observation. The parameters μ_{n} and σ_{n}^2 are estimated using the same steps as those used to estimate μ_{D} and σ_{D}^2, as mentioned above. Therefore, this model is self-adaptive and no manual setting is involved.

5.2.3. The color feature

As mentioned in Section 3.3, the illuminant-invariant image is utilized to obtain the color feature of $\mathbb{S}_{\text{int}}$. Similar to $D(S_i)$ and $N(S_i)$, a parametric probability model is built with the Gaussian parameters μ_{c} and σ_{c}^2 as follows:

$$C_{\text{prob}}(S_i) = \exp\left\{-\frac{[I_{\log}(S_i) - \mu_{\text{c}}]^2}{2\sigma_{\text{c}}^2}\right\} \tag{12}$$

where $C_{\text{prob}}(S_i)$ is the probability of S_i belonging to the drivable area, given the color observation, and $I_{\log}(S_i)$ represents the transformed pixel value of S_i.

5.2.4. The strength feature

The strength feature ($Sg(S_i)$) of S_i provides a measure of the smoothness of each superpixel, which is actually the overlap of I_{DRM} with each superpixel. The probability of S_i being the drivable area is modeled as follows:

$$Sg_{\text{prob}}(S_i) = \frac{Sg(S_i) dist(S_i, \mathbf{P}_{\text{base}})}{A(S_i)} \tag{13}$$

where $A(S_i)$ is the area of S_i and $dist(S_i, \mathbf{P}_{\text{base}})$ is the distance between S_i and $\mathbf{P}_{\text{base}}$ in the image coordinate.

5.3. The Bayesian framework

The fusion of features is conducted in the Bayesian framework. The objective is to find the posterior probability that a superpixel belongs to the drivable area, given the observations from the camera and laser sensor, $p(S_i = R|\mathbf{Obs})$, where **Obs** represents all the observations detailed above:

$$\mathbf{obs} = \{D(S_i), C(S_i), N(S_i), Sg(S_i) | S_i \in \mathbb{S}_{\text{int}}\} \tag{14}$$

Next, the probability maps obtained from the above features are interpreted as the prior conditional probability that a superpixel belongs to the drivable area. Assuming that the superpixels are conditionally independent, the posterior probability of each superpixel is obtained as follows:

$$p(S_i = R|\mathbf{Obs}) \propto \prod_{j=1}^{4} p(S_i = R|\mathbf{Obs}_j) p(S_i = R) \tag{15}$$

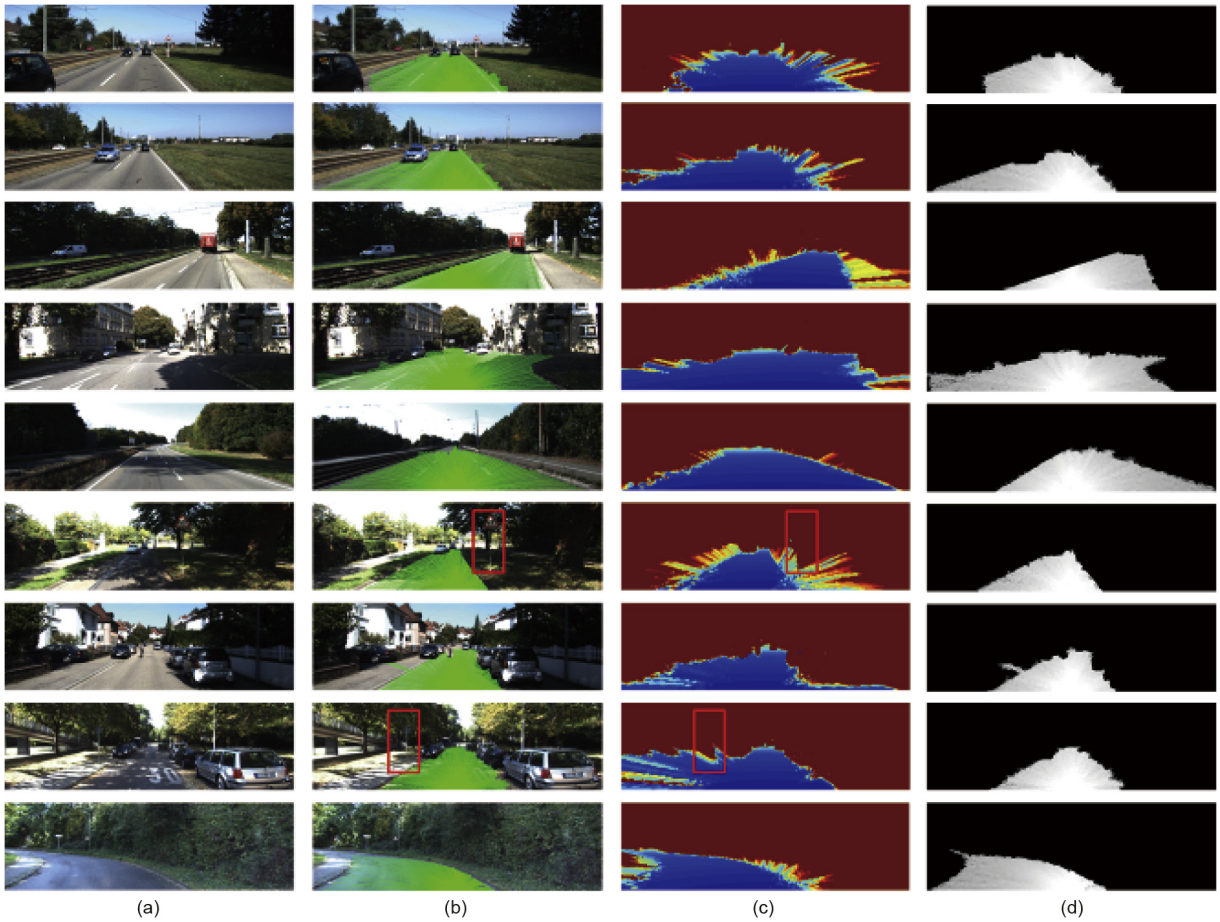

Fig. 10. The results of I_{DRM}, the DD feature, and the final probability map. Column (a) shows the original images, which contain different scenarios such as car occlusion, heavy shadow, and shifting of the road's vanishing point. Column (b) shows the corresponding I_{DRM}, which cover the drivable area well enough to generate the $\mathbb{S}_{int}$. Column (c) visualizes the proposed DD feature using Ref. [44]. The bluer an area is, the more drivable it is. The road area is very blue and the sidewalk is between blue and red, whereas obstacles such as cars and barriers are red. Column (d) shows the final probability maps, demonstrating that the proposed method performs well in different scenarios. The red bounding boxes in the sixth and eighth rows illustrate how well our method deals with small targets (such as light poles within the bounding boxes in the second column). The DD is high in front of the light poles, but there is a sudden drop in DD in the area behind these light poles, indicating that these small targets have been successfully detected.

where $p(S_i = R)$ is the probability of a superpixel belonging to the drivable area, and is obtained by averaging all $\mathbb{S}_{int}$ across the image set.

6. Experimental results and discussion

To test our proposed method, experiments were carried out using the ROAD-KITTI benchmark, which includes 289 training images and 290 testing images [5]. All experimental results were evaluated in bird's eye view (BEV) with the following metrics: max *F*-measure (MaxF), average precision (AP), precision (PRE) and recall (REC), false positive rate (FPR), and false negative rate (FNR). Three datasets were used: Urban Marked (UM), Urban Multiple Marked (UMM), and Urban Unmarked (UU). To demonstrate the efficiency of our proposed approach (see "Our method with co-point" in the tables below), we compared it with the top three laser-sensing methods from the ROAD-KITTI benchmark's website: HybridCRF, MixedCRF, and LidarHisto. We also listed the result from Ref. [36] (RES3D-Velo) below, since the data fusion procedures our method adopts are similar to those used in that reference. To show the impact that results from eliminating laser points by means of co-point mapping, the performances of our method with and without co-point mapping were compared. Moreover, we listed the performance of our method with co-point mapping on a training set, in order to demonstrate that our method is unsupervised.

As shown in Tables 1–4, our method yields the best performances in PRE and FPR in the UMM and UU datasets, which

Table 1
A comparison of our approach with the top three laser-sensing methods using the UM (BEV) dataset.

Method	MaxF	AP	PRE	REC	FPR	FNR
HybridCRF	90.99%	85.26%	90.65%	91.33%	4.29%	8.67%
MixedCRF	90.83%	83.84%	89.09%	92.64%	5.17%	7.36%
LidarHisto	89.87%	83.03%	91.28%	88.49%	3.85%	11.51%
RES3D-Velo	83.81%	73.95%	78.56%	89.80%	11.16%	10.20%
Our method with co-point	83.94%	84.83%	79.11%	89.40%	10.76%	10.60%
Our method without co-point	84.82%	85.37%	78.61%	92.09%	11.42%	7.91%
Our method on training set	85.87%	87.85%	81.56%	90.67%	9.43%	9.33%

Table 2
A comparison of our approach with the top three laser-sensing methods using the UMM (BEV) dataset.

Method	MaxF	AP	PRE	REC	FPR	FNR
HybridCRF	91.95%	86.44%	94.01%	89.98%	6.30%	10.02%
MixedCRF	92.29%	90.06%	93.83%	90.80%	6.56%	9.20%
LidarHisto	93.32%	93.19%	95.39%	91.34%	4.85%	8.66%
RES3D-Velo	90.60%	85.38%	85.96%	95.78%	17.20%	4.22%
Our method with co-point	91.80%	92.09%	91.31%	92.29%	9.65%	7.71%
Our method without co-point	92.41%	92.31%	91.61%	93.22%	9.38%	6.78%
Our method on training set	92.15%	93.56%	92.84%	91.48%	7.70%	8.52%

Table 3
A comparison of our approach with the top three laser-sensing methods using the UU (BEV) dataset.

Method	MaxF	AP	PRE	REC	FPR	FNR
HybridCRF	88.53%	80.79%	86.41%	90.76%	4.65%	9.24%
MixedCRF	82.79%	69.11%	79.01%	86.96%	7.53%	13.04%
LidarHisto	86.55%	81.13%	90.71%	82.75%	2.76%	17.25%
RES3D-Velo	83.63%	72.58%	77.38%	90.97%	8.67%	9.03%
Our method with co-point	82.57%	81.49%	75.54%	91.04%	9.61%	8.96%
Our method without co-point	83.09%	83.25%	75.52%	92.36%	9.76%	7.64%
Our method on training set	82.03%	82.82%	74.63%	91.05%	11.01%	8.95%

Table 4
A comparison of our approach with the top three laser-sensing methods using the averaged URBAN dataset (BEV).

Method	MaxF	AP	PRE	REC	FPR	FNR
HybridCRF	90.81%	86.01%	91.05%	90.57%	4.90%	9.43%
MixedCRF	89.46%	83.70%	88.52%	90.42%	6.46%	9.59%
LidarHisto	90.67%	84.79%	93.06%	88.41%	3.63%	11.59%
RES3D-Velo	86.58%	78.34%	82.63%	90.92%	10.53%	9.08%
Our method with co-point	87.02%	86.55%	83.49%	90.87%	9.90%	9.13%
Our method without co-point	87.70%	87.31%	83.50%	92.36%	10.06%	7.64%
Our method on training set	86.68%	89.71%	87.23%	91.78%	42.22%	8.44%

URBAN is the combination of UM, UMM and UU.

demonstrates that our method covers the road area well. Table 4 averages the performance; our method yields the best AP, PRE, and FPR, which reveals the robustness of our method in different scenarios. Compared with our method without co-point mapping, our method with co-point mapping eliminates around 30% of the laser data (as shown in Table 5), yet yields similar performances; this finding indicates that co-point mapping can successfully preserve the sematic structure of an image. Above all, although our method is unsupervised, it yields a competitive performance when compared with supervised methods.

Table 5
Number of laser points used by our methods.

	Our method with co-point	Our method without co-point	Eliminating rate
Training set	13 194	18 873	30.09%
Testing set	12 902	18 663	30.87%

In addition, in order to verify how much the feature fusion improves the performance, we compared the feature fusion results with each individual feature's probability map, as well as with $\mathbb{S}_{\text{int}}$ and the baseline, in Tables 6–9. All these experimental results were obtained using the training set. As Tables 6–9 show, significant improvement was achieved in MaxF and AP by means of feature fusion.

When compared with the method that used all the laser points (see "Fusion without co-point" in the tables below), the method with co-point mapping cut off around 30% of the laser points, on average, in the training set (as shown in Table 5), but still yielded a similar performance.

$\mathbb{S}_{\text{int}}$ shows exceptional performance in REC and FNR with a similar FPR as "Baseline" in the tables below, so it is a feasible choice to use $\mathbb{S}_{\text{int}}$ to estimate the parameters, as detailed in Section 5.

As shown in Fig. 11, the image areas bounded by red lines are not road areas in the ground truth in the ROAD-KITTI dataset; however, our method tends to identify these areas as foreground, since

Table 6
Ablation study of our method on UM training set (BEV).

Item compared	MaxF	AP	PRE	REC	FPR	FNR
Baseline	81.34%	86.42%	77.09%	86.09%	11.76%	13.91%
Initial drivable area	78.61%	64.00%	67.25%	94.58%	21.18%	5.42%
Color feature	83.97%	78.65%	77.57%	91.52%	12.17%	8.48%
Strength feature	81.39%	84.44%	80.00%	82.82%	9.52%	17.18%
DD feature	84.56%	74.42%	78.71%	91.34%	11.36%	8.66%
NV feature	85.38%	77.10%	81.81%	89.28%	9.13%	10.72%
Fusion with co-point	85.87%	87.85%	81.56%	90.67%	9.43%	9.33%
Fusion without co-point	86.55%	88.10%	81.90%	91.76%	9.33%	8.24%

Table 7
Ablation study of our method on UMM training set (BEV).

Item compared	MaxF	AP	PRE	REC	FPR	FNR
Baseline	79.48%	84.04%	72.83%	87.47%	35.59%	12.53%
Initial drivable area	88.06%	79.48%	82.21%	94.81%	22.38%	5.19%
Color feature	91.41%	89.92%	89.96%	92.90%	11.31%	7.10%
Strength feature	88.15%	90.71%	83.72%	93.08%	19.74%	6.92%
DD feature	90.75%	86.23%	89.64%	91.89%	11.58%	8.11%
NV feature	91.83%	89.33%	93.05%	90.64%	7.38%	9.36%
Fusion with co-point	92.15%	93.56%	92.84%	91.48%	7.70%	8.52%
Fusion without co-point	92.99%	93.81%	93.11%	92.87%	7.49%	7.13%

Table 8
Ablation study of our method on UU training set (BEV).

Item compared	MaxF	AP	PRE	REC	FPR	FNR
Baseline	73.02%	76.96%	68.57%	78.10%	12.74%	21.90%
Initial drivable area	74.26%	57.49%	60.61%	95.85%	22.16%	4.15%
Color feature	78.37%	69.23%	68.64%	91.31%	14.84%	8.69%
Strength feature	78.62%	81.54%	74.39%	83.37%	10.21%	16.63%
DD feature	81.00%	68.05%	72.23%	92.19%	12.61%	7.81%
NV feature	81.23%	69.53%	73.85%	90.24%	11.37%	9.76%
Fusion with co-point	82.03%	82.82%	74.63%	91.05%	11.01%	8.95%
Fusion without co-point	82.99%	83.36%	75.80%	91.69%	10.41%	8.31%

Table 9
Ablation study of our method on URBAN training set (BEV).

Item compared	MaxF	AP	PRE	REC	FPR	FNR
Baseline	77.95%	82.47%	72.83%	83.88%	20.03%	16.11%
Initial drivable area	80.31%	66.99%	70.02%	95.08%	21.91%	4.92%
Color feature	84.58%	79.27%	78.72%	91.91%	12.77%	8.09%
Strength feature	82.72%	85.56%	79.37%	86.42%	13.16%	13.58%
DD feature	85.44%	76.23%	80.19%	91.81%	11.85%	8.19%
NV feature	86.15%	78.65%	82.90%	90.05%	9.29%	9.95%
Fusion with co-point	86.68%	88.08%	83.01%	91.07%	9.38%	8.93%
Fusion without co-point	87.51%	88.42%	83.60%	92.11%	9.08%	7.89%

Fig. 11. Examples of areas with ambiguous semantics. The image areas bounded by red lines are not "road" in a narrow sense, but many of these areas are designed for driving.

detecting drivable areas is our main concern. Thus, the FPR value of our results is higher in the situation shown in Fig. 11. In fact, these bounded areas have ambiguous semantics, and include the transition zone between a sidewalk and road, the entrance of parking lots, and slopes for vehicles to drive onto the sidewalk. Many of these areas are designed for vehicles to drive on when necessary for convenience. In real-life application, self-driving cars are expected to choose these kinds of flat areas as candidate roads in an emergency (such as when avoiding a suddenly turning vehicle); therefore, these areas should be considered by planning algorithms.

7. Conclusion and future work

This paper proposes a self-adaptive method for drivable area detection by fusing pixel information with spatial information from laser points based on co-point mapping. Four features (the DD, NV, color, and strength features) are fused in a Bayesian framework. This method, which is based on data fusion, overcomes the disadvantages of using a single sensor when dealing with highly random and complex urban traffic scenes. Our method requires no strong hypothesis, training process, or labeled data. In addition, experiments conducted using the ROAD-KITTI benchmark testify to the efficiency and robustness of our method. Regarding future work, the first task is to divide the road into drivable areas and drivable area candidates for emergency use. Next, a dataset that can better deal with the problem of ambiguous semantics is needed. Finally, a field-programmable gate array (FPGA) implementation of our method is required in order to realize the real-time application of this method for self-driving cars.

Acknowledgements

This research was partially supported by the National Natural Science Foundation of China (61773312), the National Key Research and Development Plan (2017YFC0803905), and the Program of Introducing Talents of Discipline to University (B13043).

Compliance with ethics guidelines

Ziyi Liu, Siyu Yu, and Nanning Zheng declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Bar Hillel A, Lerner R, Levi D, Raz G. Recent progress in road and lane detection: a survey. Mach Vis Appl 2014;25(3):727–45.
[2] Zhang G, Zheng N, Cui D, Yang G. An efficient road detection method in noisy urban environment. In: 2009 IEEE Intelligent Vehicles Symposium Proceedings; 2009. P. 556–561.
[3] Cong Y, Peng JJ, Sun J, Zhu LL, Tang YD. V-disparity based UGV obstacle detection in rough outdoor terrain. Acta Autom Sin 2010;36(5):667–73.
[4] Lee DT, Schachter BJ. Two algorithms for constructing a delaunay triangulation. Int J Comput Inf Sci 1980;9(3):219–42.
[5] Fritsch J, Kuehnl T, Geiger A. A new performance measure and evaluation benchmark for road detection algorithms. In: Proceedings of International Conference on Intelligent Transportation Systems (ITSC); 2013 Oct 6–9; The Hague, the Netherlands. New York: IEEE; 2013.
[6] Tan C, Hong T, Chang T, Shneier M. Color model-based real-time learning for road following. In: Proceedings of Intelligent Transportation Systems Conference. New York: IEEE; 2006. p. 939–44.
[7] Rotaru C, Graf T, Zhang J. Color image segmentation in HSI space for automotive applications. J Real-Time Image Process 2008;3(4):311–22.
[8] Jau UL, Teh CS, Ng GW. A comparison of RGB and HSI colour segmentation in real-time video images: a preliminary study on road sign detection. In: Proceedings of the 2008 International Symposium on Information Technology; 2008 Aug 26–28; Kuala Lumpur, Malaysia. New York: IEEE; 2008.
[9] Finlayson GD, Hordley SD, Lu C, Drew MS. On the removal of shadows from images. IEEE Trans Pattern Anal Mach Intell 2006;28(1):59–68.
[10] Maddern W, Stewart A, McManus C, Upcroft B, Churchill W, Newman P. Illumination invariant imaging: applications in robust vision-based localisation, mapping and classification for autonomous vehicles. Proceedings of the Visual Place Recognition in Changing Environments Workshop; 2014 Dec 5–10; Hong Kong, China. New York: IEEE; 2014.
[11] Alvarez JM, Gevers T, LeCun Y, Lopez AM. Road scene segmentation from a single image. In: Proceedings of the 12th European Conference on Computer Vision: Volume Part VII; 2012 Oct 7–13; Florence, Italy. Heidelberg: Springer-Verlag Berlin; 2012. p. 376–89.
[12] Teichmann M, Weber M, Zoellner M, Cipolla B, Urtasun R. MultiNet: real-time joint semantic reasoning for autonomous driving. 2016. arXiv:1612.07695.
[13] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 2017;64(6):64–90.
[14] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014. arXiv:1409.1556.
[15] Badrinarayanan V, Handa A, Cipolla R. Segnet: a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. 2015. arXiv:1505.07293.
[16] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015 Jun 7–12; Boston, MA, USA. New York: IEEE; 2015. p. 3431–40.
[17] Cong Y, Liu J, Yuan J, Luo J. Self-supervised online metric learning with low rank constraint for scene categorization. IEEE Trans Image Process 2013;22(8):3179–91.
[18] Cong Y, Liu J, Fan B, Zeng P, Yu H, Luo J. Online similarity learning for big data with overfitting. IEEE Trans Big Data 2017;4(1):78–89.
[19] Alvarez JM. Gevers T. Lopez AM. 3D scene priors for road detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2010; 2010 Jun 13–18; San Francisco, CA, USA. New York; 2010. P. 57–64.
[20] Nan Z, Wei P, Xu L, Zheng N. Efficient lane boundary detection with spatial-temporal knowledge filtering. Sensors 2016;16(8):1276.
[21] Hoiem D, Efros AA, Hebert M. Recovering surface layout from an image. Int J Comput Vis 2007;75(1):151–72.
[22] Kong H, Audibert J, Ponce J. Vanishing point detection for road detection. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; 2009 Jun 20–25; Miami, FL, USA. New York: IEEE; 2009.
[23] Sivic J, Kaneva B, Torralba A, Avidan S, Freeman WT. Creating and exploring a large photorealistic virtual space. In: Proceedings of 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops; 2008 Jun 23–28; Anchorage, AK, USA. New York: IEEE; 2008.
[24] Montemerlo M, Becker J, Bhat S, Dahlkamp H, Dolgov D, Ettinger S, et al. Junior: the Stanford entry in the urban challenge. J Field Robot 2008;25(9):569–97.
[25] Thrun S, Montemerlo M, Dahlkamp H, Stavens D, Aron A, Diebel J, et al. Stanley: the robot that won the DARPA grand challenge. J Field Robot 2006;23(9):661–92.
[26] Urmson C, Anhalt J, Bagnell D, Baker C, Bittner R, Clark MN, et al. Autonomous driving in urban environments: boss and the urban challenge. J Field Robot 2008;25(8):425–66.
[27] Neidhart H, Sester M. Extraction of building ground plans from LiDAR data. Int Arch Photogramm Remote Sens Spat Inf Sci 2008;37(Pt 2):405–10.
[28] Hu X, Rodriguez FSA, Gepperth A. A multi-modal system for road detection and segmentation. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings; 2014 Jun 8–11; Dearborn, MI, USA. New York: IEEE; 2014. p. 1365–70.
[29] Fischler MA, Bolles RC. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 1981;24(6):381–95.
[30] Wellington C, Courville A, Stentz AT. A generative model of terrain for autonomous navigation in vegetation. Int J Robot Res 2006;25(12):1287–304.
[31] Klasing K, Wollherr D, Buss M. Realtime segmentation of range data using continuous nearest neighbors. In: Proceedings of 2009 IEEE International Conference on Robotics and Automation; 2008 May 13–19; Kobe, Japan. New York: IEEE; 2009. p. 2431–6.
[32] Diebel J, Thrun S. An application of Markov random fields to range sensing. Adv Neural Inf Process Syst 2005;18:291–8.
[33] Shotton J, Winn J, Rother C, Criminisi A. Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Proceedings of the 9th European Conference on Computer Vision: Volume Part I; 2006 May 7–13; Graz, Austria. Berlin: Springer; 2006. p. 1–15.
[34] Xiao L, Dai B, Liu D, Tingbo H, Tao W. CRF based road detection with multi-sensor fusion. In: Proceedings of 2015 IEEE Intelligent Vehicles Symposium (IV); 2015 Jun 28–Jul 1; Seoul, Korea. New York: IEEE; 2015. p. 192–8.

[35] Huang W, Gong X, Yang MY. Joint object segmentation and depth upsampling. IEEE Signal Process Lett 2015;22(2):192–6.
[36] Alvarez AJM, Lopez AM. Road detection based on illuminant invariance. IEEE Trans Intell Transp Syst 2011;12(1):184–93.
[37] Shinzato PY, Wolf DF, Stiller C. Road terrain detection: avoiding common obstacle detection assumptions using sensor fusion. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings; 2014 Jun 8–11; Dearvorn, MI, USA. New York: IEEE; 2014. p. 687–92.
[38] Ren X, Malik J. Learning a classification model for segmentation. In: Proceedings of the 9th IEEE International Conference on Computer Vision; 2003 Oct 13–16; Nice, France. New York: IEEE; 2003. p. 10–7.
[39] Dollár P, Zitnick CL. Structured forests for fast edge detection. In: Proceedings of 2013 IEEE International Conference on Computer Vision; 2013 Dec 1–8; Sydney, NSW, Australia. New York: IEEE; 2013. p. 1841–8.
[40] Zitnick CL, Dollár P. Edge boxes: locating object proposals from edges. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T, editors. Computer Vision—ECCV 2014; 2014 Sep 6–12; Zurich, Switzerland. Cham: Springer; 2014. p. 391–405.
[41] Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S. SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans Pattern Anal Mach Intell 2012;34(11):2274–82.
[42] Geiger A, Lenz P, Stiller C, Urtasun R. Vision meets robotics: the KITTI dataset. Int J Robot Res 2013;32(11):1231–7.
[43] Hu X, Rodriguez FSA, Gepperth A. A multi-modal system for road detection and segmentation. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings. New York: IEEE; 2014. p. 1365–70.
[44] Wang T, Zheng N, Xin J, Ma Z. Integrating millimeter wave radar with a monocular vision sensor for on-road obstacle detection applications. Sensors 2011;11(9):8992–9008.
[45] Liu J, Gong X. Guided depth enhancement via anisotropic diffusion. In: Proceedings of the 14th Pacific-Rim Conference on Multimedia; 2013 Dec 13–16; Nanjing, China. Berlin: Springer; 2013. p. 408–17.