Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

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

Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting

Jianwei Niu [a], Jie Lu [a], Mingliang Xu [b,*], Pei Lv [b], Xiaoke Zhao [a]

[a] State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China
[b] School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China

ARTICLE INFO

Article history:
Received 29 July 2015
Received in revised form
18 September 2015
Accepted 11 December 2015

Keywords:
Lane detection
Hough Transform
Cluster
Curve fitting

ABSTRACT

With the increase in the number of vehicles, many intelligent systems have been developed to help drivers to drive safely. Lane detection is a crucial element of any driver assistance system. At present, researchers working on lane detection are confronted with several major challenges, such as attaining robustness to inconsistencies in lighting and background clutter. To address these issues in this work, we propose a method named Lane Detection with Two-stage Feature Extraction (LDTFE) to detect lanes, whereby each lane has two boundaries. To enhance robustness, we take lane boundary as collection of small line segments. In our approach, we apply a modified HT (Hough Transform) to extract small line segments of the lane contour, which are then divided into clusters by using the DBSCAN (Density Based Spatial Clustering of Applications with Noise) clustering algorithm. Then, we can identify the lanes by curve fitting. The experimental results demonstrate that our modified HT works better for LDTFE than LSD (Line Segment Detector). Through extensive experiments, we demonstrate the outstanding performance of our method on the challenging dataset of road images compared with state-of-the-art lane-detection methods.

1. Introduction

In the modern society, transportation has become a vital part of daily life. As a consequence, over the past few decades, the number of vehicles in the world has increased exponentially. One negative aspect of this growth is the traffic accidents that take many lives everyday. Fatigue and ineptness are the major causes of these mishaps. Many Intelligent Transportation Systems (ITSs), such as Advanced Driver Assistance Systems (ADASs), have been developed to ensure road safety. ITS is an active research area, including tasks like obstacle detection, lane departure warning and collision prevention.

The lane is a very important part of carriageways and highways as many traffic rules for controlling and guiding drivers and reducing traffic conflicts are based on the lane. Therefore, lane detection plays a vital role in improving the performance of ITS. It can be used in a lane alarming system to warn drivers that their cars may deviate from the lane. Moreover, it can also be applied in automated driving systems. In [1], a stereo vision-based hardware and software architecture named GOLD is proposed to increase road safety.

The objective of lane detection is to separate lane markings from background clutter and to locate their exact position by employing special hardware devices or machine vision based techniques. In this work, we focus on vision-based approaches as they are cheaper in cost but can yield high detection accuracy. The lane detection algorithm should be capable of adapting to the natural outdoor surroundings, including inconsistencies in lighting, background clutter and lane occlusion.

According to [2], there are two types of approaches used for lane detection: the feature-based methods and the model-based methods. The feature-based methods are usually applied to localize the lanes in the road images by extracting low-level features. On the other hand, the model-based methods use several geometrical elements to describe the lanes, including parabolic curves, hyperbola and straight lines. Feature-based methods require a dataset containing several thousand images of the roads with well-painted and prominent lane markings that are subsequently converted to features. Moreover, these methods may suffer from noise. To avoid these issues, we opt for a model-based method. In the literature, several model-based lane detection methods are present.

We propose a novel and robust method named LDTFE to locate the lane position for an assortment of outdoor environments. First,

* Corresponding author.
E-mail addresses: niujianwei@buaa.edu.cn (J. Niu), lu_jiee@163.com (J. Lu), iexumingliang@zzu.edu.cn (M. Xu), ielvpei@zzu.edu.cn (P. Lv), zhaoke001@126.com (X. Zhao).

http://dx.doi.org/10.1016/j.patcog.2015.12.010
0031-3203/

we obtain a set of points by our proposed two-stage feature extraction. Then, we will identify final lanes using curve fitting. Our major contributions via this work are (1) we consider a lane boundary as a collection of small line segments, which can be detected more robustly against inconsistencies in lighting, weather, background clutter and lane occlusion with our proposed two-stage feature extraction; (2) we propose a modified version of Hough Transform (HT) [3] for detecting small line segments, which is capable of detecting small line segments located on a straight line or a line having small curvatures, thus providing robustness against noise; (3) we efficiently cluster the detected small line segments located on the lane boundaries by using the DBSCAN (Density Based Spatial Clustering of Applications with Noise) clustering method [4], and obtain cluster features to provide a set of points for curve fitting.

The remaining sections of this paper are organized as follows. Section 2 surveys the existing literature in this area. Section 3 presents the overview of our method. In Section 4, we describe our proposed two-stage feature extraction (includes small line segment detection and small line segment clustering). In Section 5, we describe the process of identifying the final lanes. In Section 6, the experimental results with analysis are presented. Lastly, deductions are given in Section 7.

2. Related work

Many model-based algorithms for lane detection have been proposed recently. There are four major phases in the model-based procedure: lane feature extraction, noise reduction, model fitting, and lane generation. It is important to mention that usually noise reduction is crucial for these phases, as any phase with large noise can degrade the final result irreversibly. After the phase of model fitting, sets of lane candidates are generated with a lot of false negatives. So the lane generation phase strives to handle these failure detections in an appropriate way.

Generally, there are two kinds of features for the lane detection task, i.e. colors [5–7] and edges [8–11]. As lane markings are painted in bright white or yellow on the road (obviously different in color from other parts of the road), it makes sense to consider the color feature for lane detection. Sun et al. [5] propose a method named HSILMD (HSI color model based Lane-Marking Detection), which makes use of the HSI color space. In [7], the authors propose a method for extracting lane-mark colors designed in a way that is not affected by illumination changes and the proportion of space that vehicles on the road occupy. However, most color models are sensitive to the varying illumination. Hence edge based features promise more robustness against changing light conditions. There are many conventional approaches based on edge features. In [8], a modified Canny detector with low threshold from Inverse Perspective Mapping (IPM) is applied to extract edge points. Wang et al. [9] propose a novel approach for the lane feature extraction, which is based on the observation that upon zooming into the discontinuities in the lane markings, the lane markings move on the same straight line that they are on, while other objects don't possess this characteristic. Wang et al. [10] propose a novel method to detect the edges by dividing the original image into horizontal strips, and then using contextual information for detecting the vanishing point of the lanes. In addition, Wang and Lin [12,13] propose a reconfigurable model for recognizing objects.

For model fitting, the lane is represented by using a mathematical model and then the model parameters are estimated. HT is a common approach to detect straight lines in a single binary image. In [14], a linear model based on the randomized HT is employed to detect the lanes, which is faster and more thrifty in internal storage as compared with the traditional HT. Yoo et al. [15] present a piecewise linear model based on HT. HT also has been employed in some non-linear models. For instance, a parabolic model is applied to describe the representation of the lane [16], which brings HT into play for detecting the initial lane boundary. In order to improve the representation of lanes, several deformable models [2,17–19] have been exploited to fit feature points. Wang et al. [2] suggest a novel spline-based lane model, which makes use of the cubic B-spline to fit the middle line of two sides of the lane. In their approach, two sides of the lane are assumed to be parallel. Eidehall and Gustafsson [17] present an innovative approach based on an approximation clothoid curve to estimate the road shape automatically. A hyperbola-pair model of lane boundary is introduced in [19], which detects the lane as two parallel hyperbolas. A lane detection approach based on a combination of Catmull–Rom spline with the extended Kalman filter tracking is proposed in [20], which exploits the robustness and stability of the extended Kalman filter. In [21], a lane detection algorithm is given, which utilizes gabor filters and a lane geometrical model consisting of four parameters (starting position, lane's original orientation, lane's width and lane's curvature). The work in [22] proposes lane-mark extraction by analyzing the boundaries of Regions Of Interest (ROIs) in the images and dividing the boundary images into sub-images to calculate the local edge-orientation of each block and remove bad edges. This is followed by multi-adaptive thresholding and curve fitting. Wu et al. [23] use the fan-scanning method to derive the lane-boundary information, and also exploit the angular relationships of the boundaries.

3. Overview of the LDTFE

We formulate the research problem with the lane model shown in Fig. 1. In the ideal case, the two boundary lines of a lane should be parallel. We consider that the middle line of a lane is represented by a curve line. A novel approach named LDTFE (Lane Detection with Two-stage Feature Extraction) is proposed for detecting this curve line.

In LDTFE, lane boundaries is considered as a collection of small line segments. As shown in Fig. 1, S is denoted as the set of small line segments and P stands for the set of midpoints of each small line segment in S. The curve line can be described by the function defined in the following equation:

$$f(\vec{\alpha},x)=\alpha_0+\alpha_1 x+\cdots+\alpha_k x^k \tag{1}$$

where $\vec{\alpha}=(\alpha_0,\alpha_1,\ldots,\alpha_k)$ can be estimated by using the least-squares method as shown in Eqs. (2) and (3):

$$\min \frac{1}{2}\sum_{i=1}^{n}(y_i-f(\vec{\alpha},x_i))^2 \tag{2}$$

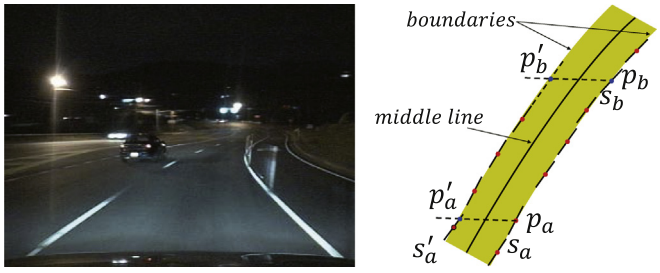

Fig. 1. A road image in night and our model for lane detection.

Please cite this article as: J. Niu, et al., Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.12.010

Here, $(x_i, y_i) \in P, i = 1, 2, \ldots, n$. Then we have

$$\begin{bmatrix} 1 & x_1 & \ldots & x_1^k \\ 1 & x_2 & \ldots & x_2^k \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \ldots & x_n^k \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \tag{3}$$

In the ideal case, the elements of the set P should be uniformly distributed on both the boundaries of the lane, which means that any point belonging to P on one side of the boundary should have its corresponding point on the other side of the boundary. However, this is not always the case. For instance, in Fig. 1, the upper part of the middle line strays towards the right boundary due to the absence of small line segments on the upper left boundary.

To solve this issue, we adopt a two-stage curve fitting method for P. The first stage involves obtaining an initial curve for P. In the second stage, we remove some noisy points and add some new points to ensure the uniform distribution of points in P on both the boundaries. Any point p should have its corresponding point p' satisfying the conditions defined in Eq. (4). As shown in Fig. 1, the point p_a on the small line segment s_a can easily find its pairing point p'_a on the line segment s'_a. All such points p_a and the mid-point of s'_a satisfy the uniform/near-uniform distribution of the elements of P. Next, consider p_b on the line segment s_b. p_b cannot find its pairing point because s_b does not have its corresponding line segment on the left boundary. So we will try to find p'_b by extending the nearest line segment (relative to s_b) located on the left boundary. If the distance $dist(p'_b, p_b)$ defined by Eq. (5) is smaller than a certain threshold, p'_b is added to the set P. If, however, $dist(p'_b, p_b)$ is larger than the defined threshold, p_b is removed from the set P. Hence we get the updated mid-point set P' which is subjected to the least-squares procedure to get the new curve line:

$$\begin{cases} f(x_i, y_i) \cdot f(x'_i, y'_i) < 0; \\ y'_i = y_i. \end{cases} \tag{4}$$

$$dist(p_i, p'_i) = |x_i - x'_i| + |y_i - y'_i| \tag{5}$$

where $p_i(x_i, y_i)$ and $p'_i(x'_i, y'_i)$ are two points.

The key problem is how to provide reliably the set of small line segments for identifying the curve line. With this issue, we proposed a two-stage feature extraction. For an input image, we obtain all small line segments from boundaries of all lanes by using the modified Hough Transform. Then, DBSCAN is applied to cluster small line segments and the noise (small line segments which do not belong the lane) are filtered out. Every cluster contains those line segments belonging to the same lane. Our lane detection framework is shown in Fig. 2.

4. Two-stage feature extraction

The proposed method starts by extracting robust and discriminative features from road images. It is essential to choose robust features. As it is difficult to extract satisfactory point sets from the results of edge detection (too much noise), we propose a feature extraction process entails two stages. First, we obtain small line segments of lane boundaries. Second, we apply a clustering algorithm to divide the small line segments into different groups. Both the stages are designed so that redundant and noisy information is filtered out, while the critical information is retained. Next, we describe the feature extraction process in detail.

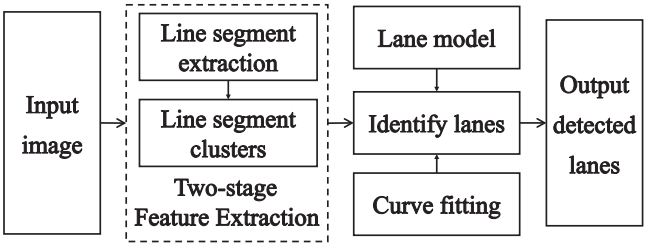

Fig. 2. The framework of our proposed lane detection approach.

4.1. Stage 1: small line segment extraction

The boundary of the lane can be regarded as a curve, whose detection poses challenges, such as: (a) the lane markings may be occluded by shadows, tyres, etc., and (b) the lane may be very short as compared to the other curves. To cater for these issues, we treat the lane as a collection of several smaller line segments. Fig. 3 (a) shows an edge image of the original road image, whereas Fig. 3 (c) consists of small line segments $s(\rho, \theta) \in S$ (the radial coordinate ρ and the angular coordinate θ), which are obtained using our modified HT.

Similar to the generalized HT, we implement HT on edge pixels, as shown in Fig. 3(a), extracted using canny algorithm. We grayed image with the efficient method proposed in [24] to enhance the character of the lane-mark information. HT transforms edge pixels defined over the image space $P(x, y)$ into the polar coordinate space $H(\rho, \theta)$ (also called parameter space), in which every point represents a particular instance of the line $l(\rho, \theta)$. The process of the transformation employs the general Eq. (6) of a straight line in polar form:

$$\rho = x \cos \theta + y \sin \theta, \quad 0^\circ < \theta < 180^\circ \tag{6}$$

where x, y is the location of the pixel and ρ, θ are the parameters of the line which passes through $p(x, y)$.

We can say that $p(x, y)$ votes to $l(\rho, \theta)$ when it satisfies Eq. (6). As shown in Fig. 3(b), three edge points vote to three series of points in polar space, respectively. After transformation, the collinear or almost collinear edge pixels generate stronger votes. The traditional HT determines lines according to the vote values. However, some nonexistent lines may have a high vote value due to the presence of many non-continuous collinear edge pixels. In this case, the detection of short lanes becomes highly improbable. Furthermore, it is difficult to choose an appropriate universal threshold because the same vote value may prove to be sufficient or insufficient for detection of a lane under different conditions. A low threshold will result in the detection of some nonexistent lines, while a high threshold will result in the loss of some important lines.

For our modified HT algorithm, we take into account the distance between adjacent edge pixels which vote for the same line. Our modified HT relies on the notion of 'continuity' (given in Definition below) which enables us to detect small line segments s located on the highly visible lanes as well as short lanes. As shown in Fig. 3, our modified HT can detect the small but important line segments, while the traditional HT fails to do so. The performance of our modified HT depends on two thresholds: TH_c and TH_l (defined in Definition below). We can optimize these two parameters in Section 6.1. We divide a line into many small line segments, which is crucial for our clustering procedure.

Definition. A line segment s consists of TH_l points belonging to the set of edge pixels $P(x, y)$. s is continuous if and only if the

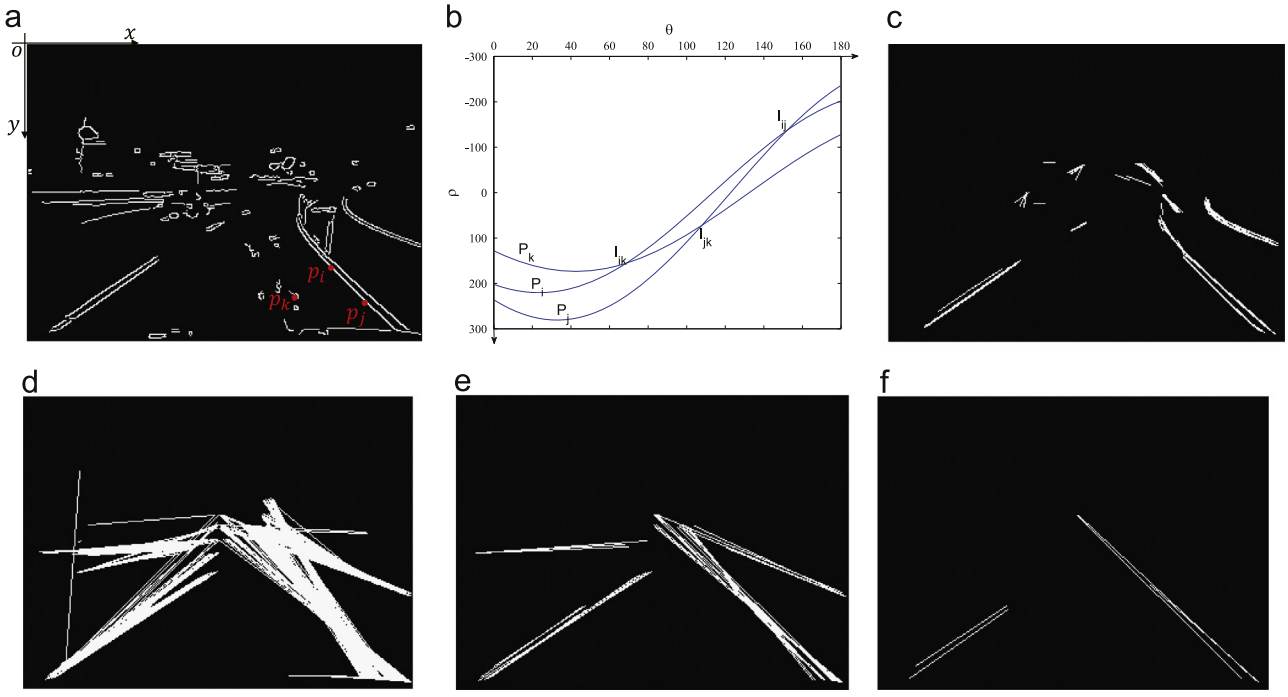

Fig. 3. (a) Three edge points $p(x_i, y_i)$, $p(x_j, y_j)$ and $p(x_k, y_k)$ are detected by the Canny algorithm. (b) The corresponding voting curves in polar space. Each intersection of the curves represents the line through the corresponding two points. (c) shows the results corresponding to our modified HT. (d), (e) and (f) show the results obtained by the traditional HT with a low, middle and high threshold, respectively.

distance (Eq. (5)) between any adjacent points ($p_i(x_i, y_i) \in s$, $p_j(x_j, y_j) \in s$ ($x_i \le x_j$) and $y_i \le y_j$) is less than TH_c.

4.2. Stage 2: small line segment clusters

As shown in Fig. 3(c), each lane corresponds to many small line segments. There are some practical issues to be dealt with: (a) the offset between the actual location and the detected location of a line segment; (b) the lane boundary is not a perfect straight line, and has small curvatures; (c) any two small line segments lying on two sides of the boundaries do not map to the same radial coordinate ρ. In addition, the presence of detection error is inevitable. These practical issues lead to the phenomenon of small line segments belonging to the same lane.

We cluster the set of the small line segments depending on the similarity measurement defined in Eq. (7). We consider mainly two aspects: (1) the degree of collinearity measured by the radial coordinate ρ and angular coordinate θ; (2) the shortest distance among small line segments. Note that the higher the similarity score between s_i and s_j is, the smaller the value $sim(s_i, s_j)$ will be:

$$sim(s_i, s_j) = \begin{cases} +\infty, & \text{if } \max\{\Delta\theta, \alpha\Delta\rho\} > \tau \\ \Delta d, & \text{others} \end{cases} \quad (7)$$

where s_i, s_j represent any two small line segments in set S (detected in Stage 1). And, $\Delta\theta = |\theta_i - \theta_j|$, $\Delta\rho = |\rho_i - \rho_j|$. ρ and θ represent the radial coordinate and the angular coordinate of the line segment. Δd indicates the shortest distance between s_i and s_j. α represents the weight of $\Delta\theta$ and $\Delta\rho$ on measuring the similarity. τ is a threshold of the degree of collinearity.

Cluster analysis is a common technique for statistical data analysis. Clustering algorithms can divide objects into various groups, or clusters. Similarity of objects within any cluster is maximized and similarity of objects belonging to different clusters is minimized. A number of clustering algorithms have been proposed. Among them, density-based algorithms can discover clusters of arbitrary shapes, whereby a cluster is defined as a connected dense component. The algorithm DBSCAN [4] is an effective density-based algorithm.

DBSCAN can determine the number of clusters automatically. Every cluster is a set of density-connected objects. During the process of clustering, noise will be filtered out effectively. We can consider that a cluster contains small line segments belonging to the same lane. DBSCAN is not suitable for the situation in which different groups have a big difference in density. This issue makes it difficult to choose the values of the parameters r and n which determine that a core object is required to contain at least n objects in its circular neighborhood of radius r. However, we are not confronted with this issue, since the small line segments belonging to a lane have densities which are uniform or nearly uniform. Therefore, in our experiments, we choose the optimum value for the parameters r and n, which can be used universally for any image showing a road with its lane markings.

As shown in Fig. 4, we obtain six clusters eventually which are divided into two groups c_{li} and c_{rj} according to the left lanes and the right lanes, respectively. Noise is filtered out effectively as compared to the set of the original small line segments (showing in Fig. 3(c)). Note that every cluster contains small line segments belonging to the same lane. Meantime noisy information is removed as much as possible.

5. Identification of lanes

We have obtained the set of clusters with the help of the straight line feature of the lane boundary. However, such feature is not enough to distinguish lanes from other objects. There are some noise clusters which have strong features of the straight line. Therefore, we identify the final lanes with two strategies: (1) find the candidate clusters which have high probability corresponding to a lane using a special characteristic of the lane; (2) identify the final lanes combining with the vanishing point.

Please cite this article as: J. Niu, et al., Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.12.010

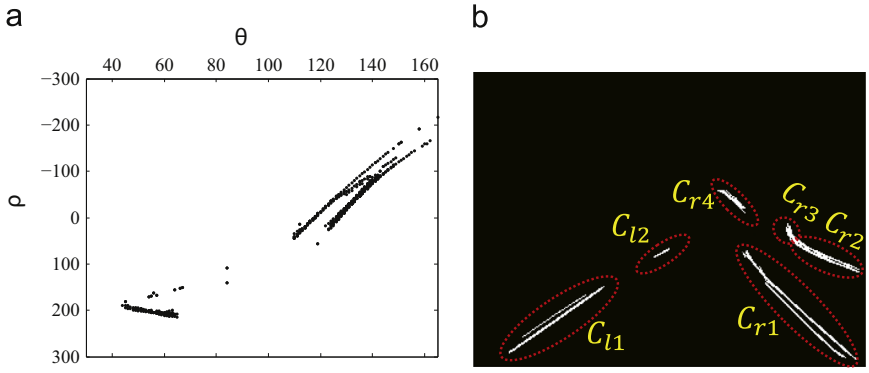

Fig. 4. (a) shows all small line segments in polar space. In particular, a point may correspond to more than one small line segments, especially in denser area. (b) gives the result of DBSCAN clustering.

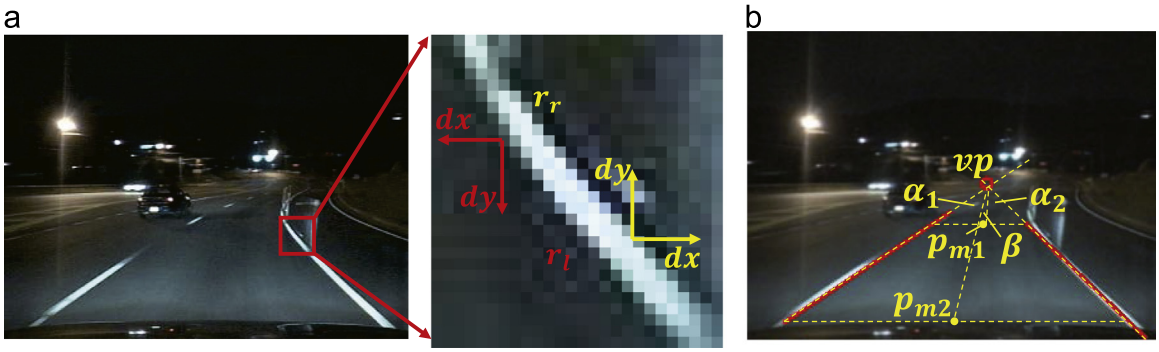

Fig. 5. (a) shows the direction of small line segments. *dx* and *dy* represent the horizontal and vertical gradients of the edge pixel, respectively. r_l and r_r are the left and the right boundaries of the right lane, respectively. (b) shows the condition what vanishing point and the lanes should satisfy.

5.1. Finding the candidate clusters

The lane itself is a strip with a certain width. Most of other objects don't have this feature. The color of the road is different from that of the lane markings. We observe that the road segments on both sides of the lane markings have the same color, and hence the same color contrast with the lane marking. This is certainly true for most parts of the lane in most situations, even after taking into account the illumination variation and the effect of shadows. We use this vital piece of information to find the candidate clusters.

The direction of a small line segment can be defined using the gradient direction of edge pixels forming the small line segment. The edge pixels on two sides of the boundaries have opposite gradient direction. We divide the gradient direction of the edge pixels into four rough directions [22] which are intervals [0,90), [90, 180), [180, 270) or [270, 360). We assign a rough direction to each small line segment relying on the most of the edge pixels direction. As shown in Fig. 5(a), we can conclude that small line segments located on the same side of the boundaries have the same direction, while two small line segments located on the opposite lane boundaries have the opposite direction. We count the number of two opposite directions in the cluster. The cluster c_i is no longer considered as a candidate when it does not satisfy the condition in the following equation:

$$w(c_i) = \frac{|n_{il} - n_{ir}|}{n_{il} + n_{ir}} < \tau_1 \tag{8}$$

where n_{il} and n_{ir} are denoted as the number of small line segments (in cluster c_i) with the opposite rough directions.

5.2. Identifying the final lanes

We employ a near exhaustive strategy in the candidate clusters because the number of the clusters is not much. Taking the left lane as an example, we get all possible curve lines for different combinations of clusters. We deal with the right lane in the same way. We can identify the final lanes with the vanishing point which can be detected by using the method described in [25]. Considering the lanes are parallel, therefore, the lanes and vanishing point should satisfy the condition: (1) the lane should pass through the vanishing point roughly; (2) $\alpha_1 - \alpha_2$ and β should tend toward zero. As shown in Fig. 5(b), α_1 and α_2 are angles between the corresponding lanes and the lines (vp, p_{m1}), (vp, p_{m2}), respectively. β is the angle between the lines (vp, p_{m1}) and (vp, p_{m2}). vp represents the vanishing point. p_{m1} and p_{m2} are middle points of the lines.

6. Experimental results

We perform experiments using the Microsoft VS 2012 environment on an Intel Core i2 2.83 GHz CPU equipped with 4 GB RAM. We create our dataset of road images by collecting images from online sources. These images cover variegated environments such as sunlight, nightfall, rainy conditions and tunnels. All images are scaled to 320×240 pixels. The images fulfill the following criteria: (a) the lane markings should be visible to the naked eye, (b) the area of the road must occupy more than 60% of the image space, and (c) no large lane area should be occluded by vehicles.

In order to carry out this experiment, we divide the dataset into two groups: (1) We pick 500 road images randomly from our original images. And, we record the extremities of the lane markings in each of these images. (2) Another dataset consists

3000 road images which are grouped into four categories. All those road images satisfy the conditions we defined earlier. The databases are available from the link below this page. We divide our experiments into three parts. Firstly, we analyze the effects of parameters on the lane detection performance. Secondly, we have a comparative performance evaluation: (1) we compare our modified HT with LSD [26] in our LDTFE method; (2) we use optimal parameters for evaluating the performance of our method on two datasets.

6.1. Finding the optimal configuration parameters

The performance of our modified HT is largely affected by two thresholds TH_c and TH_l. In addition, the DBSCAN algorithm needs to be provided with the optimum values for r and n. As a whole, all parameters interplay in two-stage feature extraction. Therefore, we need to calculate the best possible values for these variables. We try different combinations of the (TH_c, TH_l, r, n) quadruplet, and measure the corresponding performance of our detection algorithm, using the performance metrics of Mean Accuracy Rate (MAR) and Standard Deviation (SD). MAR reflects the average detection rate of the detector, while SD gives a cue about the stability of MAR.

The values of the four configuration parameters are varied between the ranges shown in Fig. 6. These ranges are chosen by analyzing some previous work in this domain. For each iteration of the experiment, three of the configuration parameters are fixed, while the forth parameter is varied from its lower bound to the upper bound. For each configuration, the corresponding MAR and SD values are determined as displayed in Fig. 6. Fig. 7 shows some examples.

It is apparent that the impact of changing the variables TH_c and TH_l is much bigger than that of changing r and n. It can also be observed that the higher the MAR value is, the lower the corresponding SD value will be. Our method exhibits an average MAR of 86.4% for the 54 configurations having TH_c, TH_l, r and n in the ranges [4,5], [7,11], [3,5] and [8,10], respectively. The highest MAR value (96.3%) is achieved for the quadruplet (4, 9, 4 and 9). Hence these are safely assumed to be the optimal configuration parameters.

6.2. Comparative performance evaluation

6.2.1. Comparison between proposed modified HT and LSD

In our approach, we propose a modified HT for detecting small line segments in Stage 1. LSD (Line Segment Detector) is a representative algorithm for line detection proposed in recently. It can be used to detect locally straight line segments in linear time. In order to serve our approach better, we segregate the road image into a few horizontal slices which overlap each other by a small amount. Then, LSD is applied to each slice. The result of the Stage 1 will be used as the input of the Stage 2. We compare the AR with different parameters in the Stage 2, as shown in Fig. 8.

It is apparent that the AR values for the modified HT algorithm are more stable with the change of n and r. This proves that the modified HT is more robust than LSD for a wide variety of road scenes. Hence it can be safely concluded that the proposed modified HT outperforms the LSD. The reason is that HT can generate more than one line segment for a real edge, which is favorable for clustering small line segments in Stage 2. Although the LSD algorithm is faster than our modified HT, the modified HT still satisfy our requirement. The average detection time of the modified HT and LSD are 11.5 ms and 8.4 ms, respectively.

6.2.2. Performance analysis under two datasets

We measure the detection performance of the LDTFE using the optimal configuration parameters on the four categories of the images in our dataset and the dataset in [27]. The experimental results are shown in Fig. 9. We evaluate the performance using the metrics of Accuracy Rate (AR) and False Negative (FN).

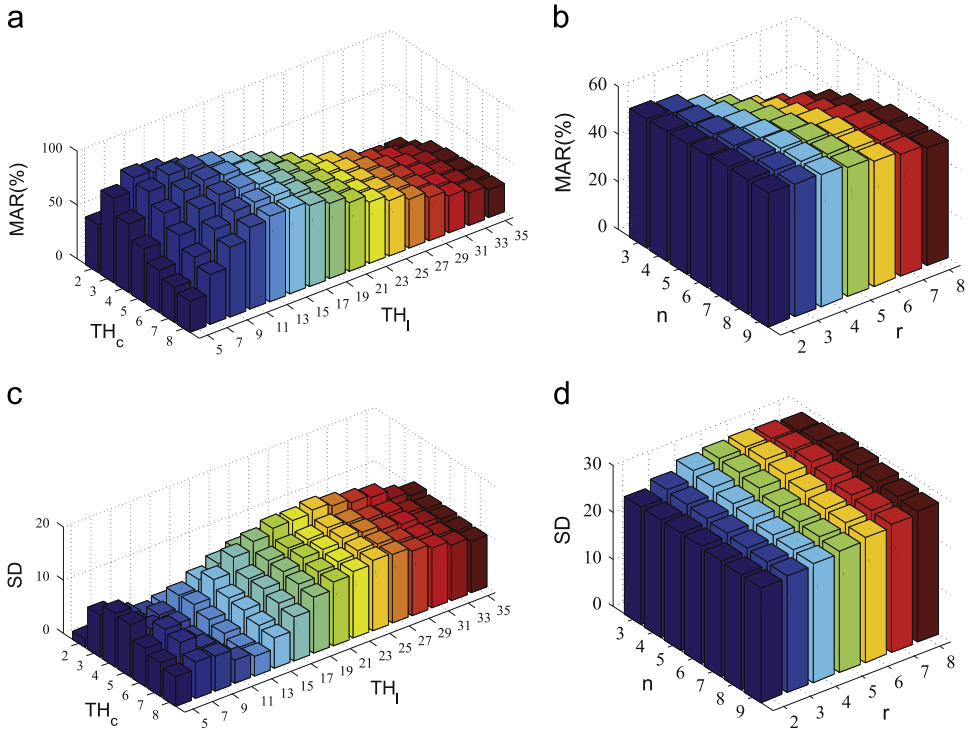

Fig. 6. (a) The MAR values for the ranges TH_c [2,8] and TH_l [5,35]. Each bar represents the value of MAR achieved by averaging the accuracy rates for the point (TH_l, TH_c) and the full ranges of r and n. (b) The MAR values for the ranges r [2,8] and n [3,9]. Each bar represents the value of MAR achieved by averaging the accuracy rates for the point (r, n) and the full ranges of TH_c and TH_l. (c) and (d) The corresponding SD values for (a) and (b).

Fig. 7. The first row shows some detection results with LDTFE applied under various conditions, such as noise from text painted nearby the lane marking, nighttime, fog, and other vehicles in close vicinity. The second row shows small line segments after clustering; the two colors represent opposite direction of small line segments.

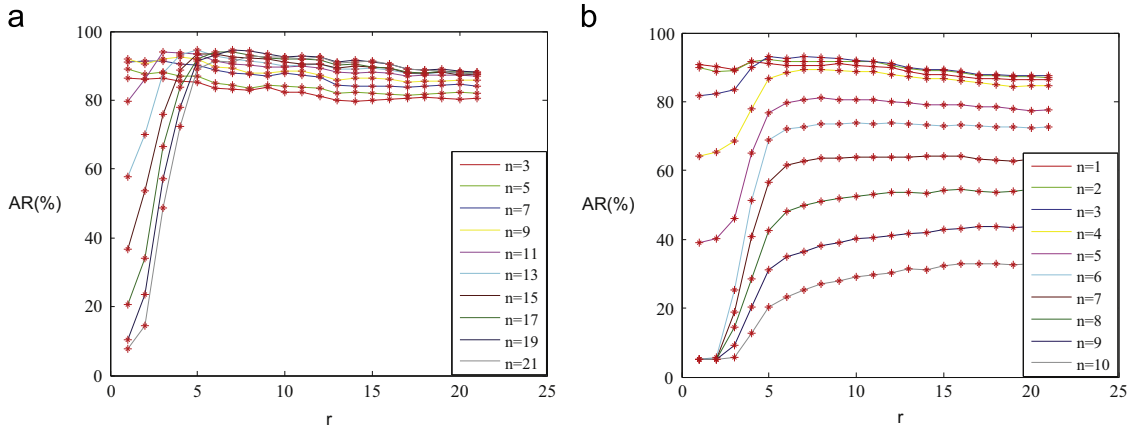

Fig. 8. (a) and (b) show the impact changing n and r on the Average Rate (AR) corresponding to the modified HT (threshold on TH_c 4 and TH_l 9) and LSD, respectively .

Fig. 9. The first two rows show the detection results from our dataset under four scenarios. The last two rows show the results corresponding to the Caltech dataset in four clips.

Our dataset: It is worth pointing out the challenges presented by the four scenarios. For the daylight images, strong ultraviolet light from the sun creates illumination inconsistencies which deteriorate the visibility. As compared to the daylight images, the lighting conditions at nighttime are quite different, giving rise to a different set of challenges. The lane markings have a low contrast with the road, and the artificial light beams emanating from the vehicle headlights change the local contrast, besides generating edges on the segments of the road that are lit up. For the rainy images, the raindrops on the vehicle's windshield blur the whole image. The tunnel images consist of reflections due to the tunnel's ceiling lights, and a drastic illumination change occurs when the vehicle approaches the end of the tunnel. The experimental results are shown in Table 1.

Caltech dataset: The dataset [27] consists of four video sequences (Cordova1, Cordova2, Washington1, Washington2) with the size of 640×480. They have different challenges, including having a lot of curvatures, writings on the road or shadows. We

Please cite this article as: J. Niu, et al., Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.12.010

Table 1
The performances of different detection methods under our dataset.

Scenario	Number	Wu's method [22]		LDTFE	
		AR (%)	FN (%)	AR (%)	FN (%)
Day-time	750	87.7	6.7	91.5	4.7
Night-time	750	92.2	4.1	95.9	1.9
Rain	750	93.5	3.4	94.8	2.1
Tunnel	750	89.6	3.6	92	2.3
Total	3000	90.8	4.45	93.6	2.8

Table 2
The performances of different detection methods under Caltech dataset.

Clip	Number	Aly's method [28]		LDTFE	
		AR (%)	FP (%)	AR (%)	FN (%)
Cordova1	250	97.2	3.0	92.2	5.4
Cordova2	406	96.2	38.4	97.7	1.8
Washington1	337	96.7	4.7	96.9	2.5
Washington2	232	95.1	2.2	98.5	1.7
Total	1225	96.3	11.6	96.5	2.7

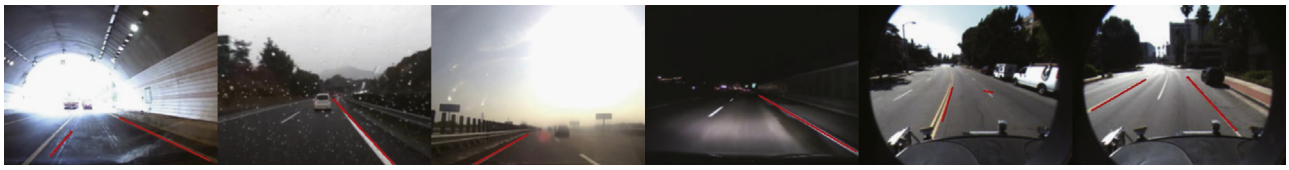

Fig. 10. Failed detection cases.

detect 2-lane, in other words, we detect only the two lanes where the vehicle stays in. The experimental results are shown in Table 2.

It can be seen that our method achieves better performances for all the four scenarios compared to Wu's method without tracking. The average detection accuracy on all the images is 93.6%. The reason is that our two-stage feature extraction exploits the relationship among edge points to help remove more noise. Certainly, it is worthwhile to note that our method is not good at those road images with the large curvature compared to Wu's method. In addition, both of the methods perform not well in the day-time scenarios, in which the image is overexposed under the strong sunlight. In Caltech dataset, our method outperforms Aly's method in overall. Our method is not good at detecting the clip of Cordova1 compared with other clips, in which the lanes have a lot of curvature. The proposed approach is bad in thus scene, as we state in the next section. Some of the failed detection cases are given in Fig. 10.

6.3. *Limitations of our approach*

We acknowledge that there are some limitations in our proposed method. First, for lanes with large curvature, our method does not generate very accurate small line segments, as it becomes more difficult to treat the local curvature as straight line. However, it is pertinent to mention here that in most of the cases, the road turns are not abrupt enough to create large curvature for lanes in the road images. Therefore, the applicability of LDTFE is not much affected. Second, as we mentioned in the experiments, we test our algorithm for images in which lane markings are not significantly occluded. One could argue that during heavy traffic, drivers tend to change lanes more frequently, thereby occluding the lane markings. We do not deny this possibility. LDTFE does not explicitly address lane occlusions. However, we believe that for reasonable occlusion, LDTFE is able to detect the final lane.

7. Conclusion

We have presented a robust method named LDTFE to detect lanes, in which we take lane boundary as collection of small line segments. We have proposed an improved variant of HT, which has the ability to detect small line segments located on a straight line or a line having small curvature. To minimize the occurrence of false positives, we identify the final lanes using the vanishing point, after finding the candidate clusters utilizing the color contrast between the road and the lane boundaries. Through our experiments, we have proved that our method gives excellent results on two datasets of road images.

Conflict of interest

None declared.

Acknowledgments

This work was supported by the 973 Program (2013CB035503), National Natural Science Foundation of China (61572060, 61170296, 61190125, 61202207 and 61472370) and the R&D Program (2013BAH35F01) of China.

References

[1] M. Bertozzi, A. Broggi, Gold: a parallel real-time stereo vision system for generic obstacle and lane detection, IEEE Trans. Image Process. 7 (1) (1998) 62–81.
[2] Y. Wang, E.K. Teoh, D. Shen, Lane detection and tracking using b-snake, Image Vis. Comput. 22 (4) (2004) 269–280.
[3] R.O. Duda, P.E. Hart, Use of the hough transformation to detect lines and curves in pictures, Commun. ACM 15 (1) (1972) 11–15.
[4] C. Cassisi, A. Ferro, R. Giugno, G. Pigola, A. Pulvirenti, Enhancing density-based clustering: parameter reduction and outlier detection, Inf. Syst. 38 (3) (2013) 317–330.
[5] T.-Y. Sun, S.-J. Tsai, V. Chan, HSI color model based lane-marking detection, in: IEEE on Intelligent Transportation Systems Conference, 2006, pp. 1168–1172.
[6] K.-Y. Chiu, S.-F. Lin, Lane detection using color-based segmentation, in: Proceedings of the IEEE Intelligent Vehicles Symposium, 2005, pp. 706–711.
[7] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, K.-C. Fan, Lane detection with moving vehicles in the traffic scenes, IEEE Trans. Intell. Transp. Syst. 7 (4) (2006) 571–582.
[8] H. Li, F. Nashashibi, Robust real-time lane detection based on lane mark segment features and general a priori knowledge, in: 2011 IEEE International Conference on Robotics and Biomimetics, pp. 812–817.
[9] Y. Wang, N. Dahnoun, A. Achim, A novel system for robust lane detection and tracking, Signal Process. 92 (2) (2012) 319–334.
[10] Y. Wang, L. Bai, M. Fairhurst, Robust road modeling and tracking using condensation, IEEE Trans. Intell. Transp. Syst. 9 (4) (2008) 570–579.
[11] R. Jiang, M. Terauchi, R. Klette, S. Wang, T. Vaudrey, Low-Level Image Processing for Lane Detection and Tracking, Arts and Technology, Springer, Taiwan, 2010.
[12] X. Wang, L. Lin, L. Huang, S. Yan, Incorporating structural alternatives and sharing into hierarchy for multiclass object recognition and detection, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3334–3341.
[13] L. Lin, X. Wang, W. Yang, J.-H. Lai, Discriminatively trained and-or graph models for object shape detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (5) (2015) 959–972.
[14] J. Wang, Y. Wu, Z. Liang, Y. Xi, Lane detection based on random hough transform on region of interesting, in: IEEE International Conference on Information and Automation, 2010, pp. 1735–1740.
[15] H. Yoo, U. Yang, K. Sohn, Gradient-enhancing conversion for illumination-robust lane detection, IEEE Trans. Intell. Transp. Syst. 14 (3) (2013) 1083–1094.
[16] J. Wang, F. Gu, C. Zhang, G. Zhang, Lane boundary detection based on parabola model, in: IEEE International Conference on Information and Automation (ICIA), 2010, pp. 1729–1734.
[17] A. Eidehall, F. Gustafsson, Obtaining reference road geometry parameters from recorded sensor data, in: IEEE on Intelligent Vehicles Symposium, 2006, pp. 256–260.
[18] J. McCall, M. Trivedi, Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation, IEEE Trans. Intell. Transp. Syst. 7 (1) (2006) 20–37.
[19] Q. Chen, H. Wang, A real-time lane detection algorithm based on a hyperbola-pair model, in: IEEE on Intelligent Vehicles Symposium, 2006, pp. 510–515.
[20] K. Zhao, M. Meuter, C. Nunn, D. Muller, S. Muller-Schneiders, J. Pauli, A novel multi-lane detection and tracking system, in: IEEE on Intelligent Vehicles Symposium (IV), 2012, pp. 1084–1089.
[21] S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, H. Chen, A novel lane detection based on geometrical model and gabor filter, in: IEEE on Intelligent Vehicles Symposium (IV), 2010, pp. 59–64.
[22] P.-C. Wu, C.-Y. Chang, C.H. Lin, Lane-mark extraction for automobiles under complex conditions, Pattern Recognit. 47 (8) (2014) 2756–2767.
[23] C.-F. Wu, C.-J. Lin, C.-Y. Lee, Applying a functional neurofuzzy network to real-time lane detection and front-vehicle distance measurement, IEEE Trans. Systems Man Cybern. Part C: Appl. Rev. 42 (4) (2012) 577–589.
[24] T.-Y. Sun, W.-C. Huang, Embedded vehicle lane-marking tracking system, in: 2009 IEEE 13th International Symposium on Consumer Electronics, pp. 627–631.
[25] M. Nieto, L. Salgado, Real-time robust estimation of vanishing points through nonlinear optimization, in: SPIE Photonics Europe, International Society for Optics and Photonics, 2010, pp. 772402–772402.
[26] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, G. Randall, LSD: a line segment detector, Image Process. On Line 2 (2012) 35–55.
[27] M. Aly ⟨http://vision.caltech.edu/malaa/datasets/caltech-lanes/⟩, January 2014, pp. 627–631.
[28] M. Aly, Real time detection of lane markers in urban streets, in: 2008 IEEE on Intelligent Vehicles Symposium, pp. 7–12.

Jianwei Niu received Ph.D. degree in computer science from Beijing University of Aeronautics and Astronautics in 2002. He is a Professor in School of Computer Science and Engineering, Beihang University. He has published more than 100 referred papers and filed more than 30 patents in mobile and pervasive computing.

Jie Lu received the B.E. degree from College of Information Science and Engineering, Shandong University of Science and Technology, China. He is now pursuing the M.E. degree from Beihang University, China. His research interests include computer graphics and computer vision.

Mingliang Xu is an Associate Professor in the School of Information Engineering of Zhengzhou University, China. His research interests include computer graphics and computer vision. Xu got his Ph.D. degree in computer science and technology from the State Key Lab of CAD&CG at Zhejiang University.

Pei Lv is an Assistant Professor in School of Information Engineering, Zhengzhou University, China. His research interests include video analysis and crowd simulation. He received his Ph.D in 2013 from the State Key Lab of CAD&CG, Zhejiang University, China.

Xiaoke Zhao received his M.S. degree in computer science in 2014 from Beihang University, China, B.S. degree in 2011 from Guiyang University. He is now a Ph.D. student in Beihang University. His current research interests include computer vision and pattern recognition.

Please cite this article as: J. Niu, et al., Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.12.010