Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **4**, July **2018**

Lane Detection Based on Ternary Tree Traversal

Guoquan Jiang, Duo Wang, Hongmin Liu*, Zhiheng Wang

School of Computer Science and Technology
Henan Polytechnic University
2001 Century Avenue, High-tech District, Jiaozuo, Henan, 454000, China
*Corresponding author: hongminliu@hpu.edu.cn

Received July, 2017; revised February, 2018

Abstract. *In view of the complexity and inaccuracy of lane detection algorithm, a new method based on ternary tree traversal was proposed. Firstly, the binary image was obtained by image preprocessing, including the adaptive selection of region of interest(ROI), image segmentation and dilation. Secondly, candidate feature points indicating the center line of lane were extracted under the constraint of distance threshold. Thirdly, to obtain their position distribution information, the candidate feature points were traversed using the ternary tree traversal; a filtering algorithm was presented to filter the pseudo feature points in the candidate feature points and get effective feature points, which was designed according to the quantity constraint of consecutive candidate feature points and the parameters constraint of its fitting line. Finally, the effective feature points were fitted by RANSAC(random sample consensus), and then the equations of lanes were obtained. In addition, in order to improve the robustness of the algorithm, especially when there is no lane in current frame, a prediction model was put forward. Test results show that the proposed method can detect lane quickly and effectively in most of the complex road conditions, such as part of lanes missing, vehicles passing and symbols on the road surface existing, etc. For a color image of 640×480 pixels, the average cost time is about 116ms, and the detection rate reaches 91%.*

Keywords: Lane detection, Ternary tree traversal, Pseudo feature points filtering, Prediction model.

1. **Introduction.** Road signs detection and road lane detection, lane departure warning(LDW) and other applications make the development of vehicle intelligent system more rapidly [1]. Road lane detection is a key technology in vehicle intelligent system architecture [2]. Because of the variability and uncertainty of the road scene, lane detection has become a challenging task. To this end, domestic and foreign scholars have done a lot of research.

Yoo et al. [3] proposed a gradient enhancement method to detect lane lines in 2013, which used a piecewise linear model based on Hough transform. Wang et al. [4] introduced a Hough transform when using the edge detector to detect the edge, and then constructed a parabolic model to describe the lane. Borkar et al. [5] used a Gaussian template matching method after using the Hough transform in order to improve lane detection rate. Peng et al. [6] proposed a lane detection algorithm based on fast random sampling consistency curve fitting verification. Xiao et al. [7] identified the lane based on Beamlet transform and K-means clustering method.

Hough transform is an effective line detection method, which is robust in lane detection, but its disadvantage is that it has high false positive rate and time complexity [2,8]. To overcome these two shortcomings, Wang et al. [9] detected lane line through random

Hough transform, which used random strategy and dynamic linked list. As a result, the algorithm's time-consuming and space complexity were reduced. However, it was still difficult to meet the real-time requirements of lane detection [10]. Li et al. [11,12] used adaptive random Hough transform to overcome these shortcomings, but there was still a high false positive rate, and it was often difficult to determine whether a straight line corresponds to the edge of a lane line.

In order to improve real-time performance and accuracy of lane detection, a method of lane detection based on ternary tree traversal and lane-line feature information is proposed. By extracting the candidate feature points, the number of pixels to be processed in pseudo-feature points filter stage is reduced and the computational efficiency of the algorithm is improved. Based on the analysis of the feature points of lanes and noises, a model of filtering pseudo-feature points is presented, which avoids the use of parameter quantization and complex mathematical operations to improve the accuracy of lane detection.

The rest of this paper is organized as follows: we first introduce image preprocessing in Section 2. Section 3 and section 4 give a detailed description of the proposed approach. Section 5 reports the experiments and results. Section 6 concludes this paper.

2. Image preprocessing.

The given image preprocessing reduces the apparent noises and enhances the contrast of the images for positive detection. We performed adaptive selection of ROI, adaptive threshold binarization and morphological operations.

2.1. Selection of the ROI.

Selecting the ROI can reduce the noise interference and improve the efficiency of the algorithm. Therefore, the ROI of the image being detected is selected in the beginning. The images used for the detection are color image with an original size of 640 × 480 and the first image rigidly selects the middle 560 × 180 area as the ROI. For the remaining images, the following adaptive ROI selection method is adopted.

Let $Col(Pi)$ and $Count(P)$ be the column coordinates and the number of the lane feature points $P(x, y)$(the detection of lane feature points is detailed in the third part), respectively. The dynamic ROI is an area extending from both sides of the center line $y = L_{mid}$, and L_{mid} can be defined as:

$$L_{mid} = \sum_{P_i(x,y)\in P(x,y)} Col(P_i(x, y))/Count(P(x, y)) \tag{1}$$

where the part of dynamic ROI, which exceeds the image, will be automatically truncated. Fig.1(b) shows the ROI, that is, the red dotted box area in Fig.1(a).

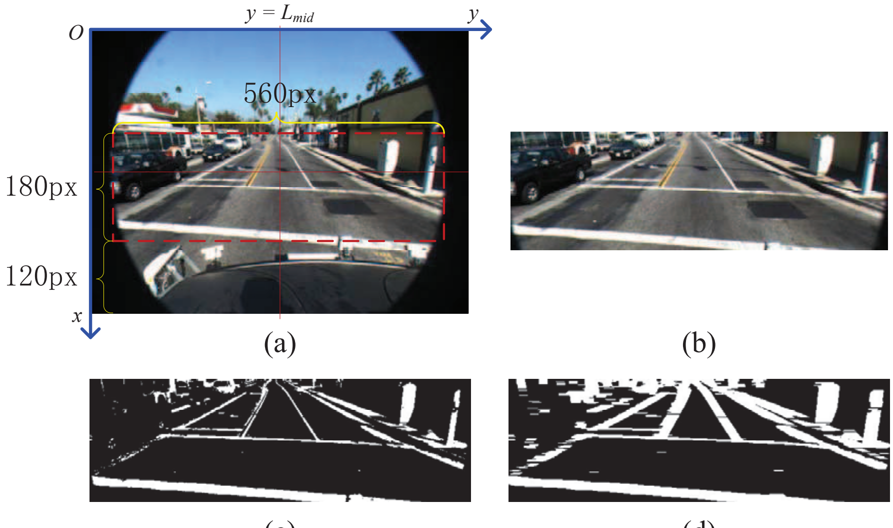
2.2. Image segmentation and dilation.

Image segmentation and dilation are performed after selecting the ROI. The purpose of segmentation is to obtain meaningful foreground pixels, and then extract the effective feature points. The selected gray scale conversion model is as follows:

$$P_{gray} = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \tag{2}$$

where R, G, B are the red, green and blue channels of the image, respectively. Then, the grayscale image is binarized using Otsu method [13] which widely used in many issues currently. Finally, the segmented image is horizontally dilated using the structural elements se shown in (3):

$$se = [1\ 1\ 1....1\ 1\ 1] \tag{3}$$



(a)

(b)

(c)

(d)

FIGURE 1. Image pre-processing. (a) original image, (b) ROI, (c) image segmentation, (d) Image dilation.

where *se* is a flat structure element with a size of 1 × 13. Fig.1(c) shows the effect of image segmentation. It can be seen from the Fig.1(c) that the lanes and some of higher brightness objects are retained, while dark pixels are filtered. Fig.1(d) shows the expanded image. As can be seen from the Fig.1(d), the lanes and noises are inflated in the horizontal direction, and most of the isolated noises are merged.

3. **Feature points detection based on ternary tree traversal.** In this section, a feature points detection method based on ternary tree traversal was proposed, which extracted the local effective feature accounting for global detection. A series of discriminant for the candidate feature points were made to obtain valid feature points.

3.1. **Candidate feature points extraction.** In order to improve the efficiency of the algorithm, it is necessary to extract candidate feature points. On the basis of the preprocessing result, the midpoint extraction algorithm based on the left and right edge points of the lane is used to obtain the candidate feature points representing the lane. Fig.2 shows a schematic representation of the candidate feature points extraction process, where the blue solid points are the extracted candidate feature points.

Candidate feature points extraction algorithm is as follows: scan the current image from left to right and from top to bottom, and set $yLeft$, $yRight$ and $yMiddle$ to store the left edge, right edge of lane or noise and the column coordinate of the candidate feature point $P(x, y)$, respectively. When the current line is scanned, the column coordinate P_L of the first foreground pixel encountered is stored in the $yLeft$. Then scanning to the right and looking for the first background pixel, its column coordinate P_R is obtained and $(P_R - 1)$ is stored in $yRight$. So the distance between the left and right edge points can be got:

$$dist = yRight - yLeft \tag{4}$$

According to the temporal consistency [1] of the lane width, the distance between the left and right edges of the lane does not change greatly. Using the constraint of distance threshold of the current frame, the candidate feature points extraction model is given as:

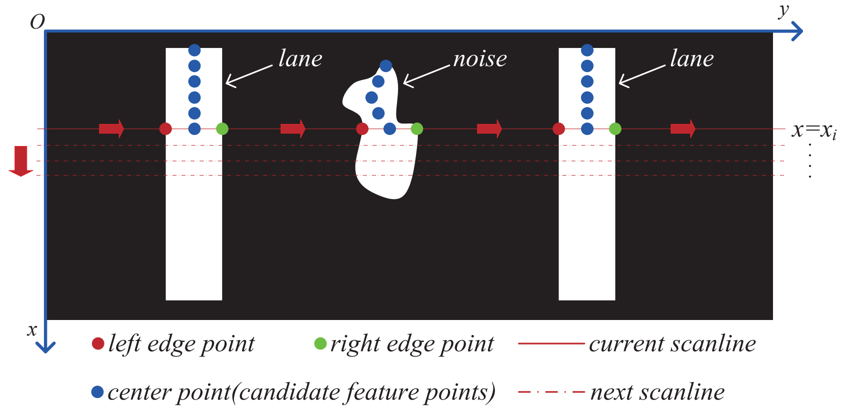


FIGURE 2. Candidate feature points extraction process

$$\lambda(P) = \begin{cases} 1, & if(\alpha_1 < dist < \alpha_2 \; and \; 0 < yLeft, yRight < \alpha_3) \\ 0, & otherwise \end{cases} \tag{5}$$

Candidate feature point is detected when $\lambda(P)$=1; otherwise, there is no candidate feature point in this time, continuing to scan from the second pixel after $yRight$ until the image is scanned completely. Generally, $[\alpha_1, \alpha_2]$ is the range of the lane width obtained based on its temporal consistency, and α_3 is the width of the ROI. The column coordinate $yMiddle$ of candidate feature point can be calculated as follows:

$$yMiddle = (yLeft + yRight)/2 \tag{6}$$

and then add the feature point to the candidate feature points set.

Fig.3 shows the results of the extraction of candidate feature points and the partial magnification of feature points at two different locations. In this figure, the feature points of the lanes and noises are extracted. In order to obtain the effective features of the lanes, we need to filter out the noises.

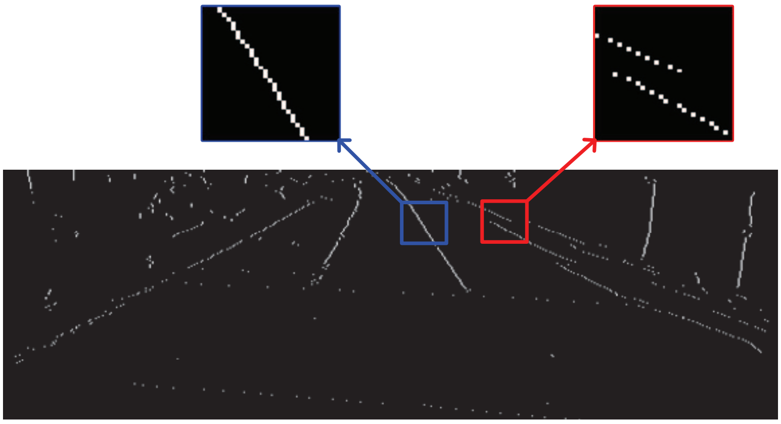


FIGURE 3. The extraction results of the candidate feature points and the partial magnification

3.2. **Pseudo-feature points filter.** In Fig.3, the candidate feature points of the lanes are discrete or continuous. The candidate feature points in the upper left enlargement are continuous because there is a large angle between the lane and the horizontal line.

On the contrary, the candidate feature points in the upper right enlargement are discrete because the angle between the lane and the horizontal line is small.

In order to filter the pseudo-feature points from the candidate feature points, a series of consecutive candidate feature points is called pixels bar $B(x, y)$(a single candidate feature point is seen as a special case of pixels bar). Considering the presence of vertical pixels bar, each pixels bar is treated as a ternary tree instead of a binary tree. Fig.3 shows that a ternary tree, which corresponds to a lane, is usually only one sub-tree, that is, the left or right lane is generally only left or right sub-tree, respectively. Also, the slope value k of the lane fitting line is always within a limited interval $[-k_{max}, k_{max}]$, and Fig.4 vividly illustrates this reason. Similarly, the intercept b of the lane fitting line should also be within a certain distance from the centerline $y = L_{mid}$ of ROI. The number P_{num} of feature points in a pixels bar :

$$P_{num} = \sum_{P(x,y)\in B(x,y)} 1 \tag{7}$$

should be satisfied the constraint of quantity. The detection model of the pixels bars belonging to the lanes is proposed as:

$$\lambda(B) = \begin{cases} 1, & if(|k| < k_{max}, k_L < 0, k_R > 0 \; and \; P_{num} > \alpha_4 \; and \; |b - L_{mid}| < \alpha_5) \\ 0, & otherwise \end{cases} \tag{8}$$

where $\lambda(B) = 1$ denotes that the pixels bar $B(x, y)$ at the current frame belongs to a lane, and k_L and k_R represent the slope values of the fitting lines of left or right sub-tree, respectively. And α_4 and α_5 are the quantity and intercept threshold, respectively. The rule in Eq.(8) is built based on the observation that the pixels bars belonging to a lane are continuous and have a certain number of feature points, and its slope values are within a certain interval.

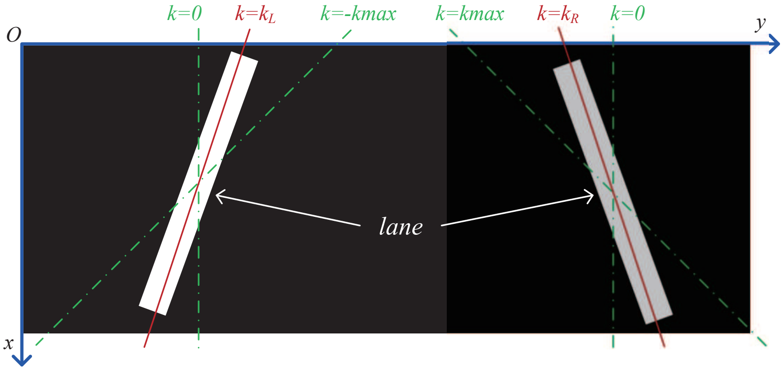


Figure 4. Slope range of the fitting line of lanes

All ternary trees are traversed sequentially. The pseudo-feature points are filtered during the traversal process. Fig.5 shows a flow chart of pseudo-feature points filtering. The implementation process of the pseudo-feature points filtering is described in detail below.

Step 1: Traverse the feature points in the pixels bar $B(x, y)$ and count, then preserve the coordinates of each feature point into the position matrix Pos: $Pos = \{(x, y)|p(x, y) \in B(x, y)\}$,where the size of the Pos is $P_{num} \times 2$.

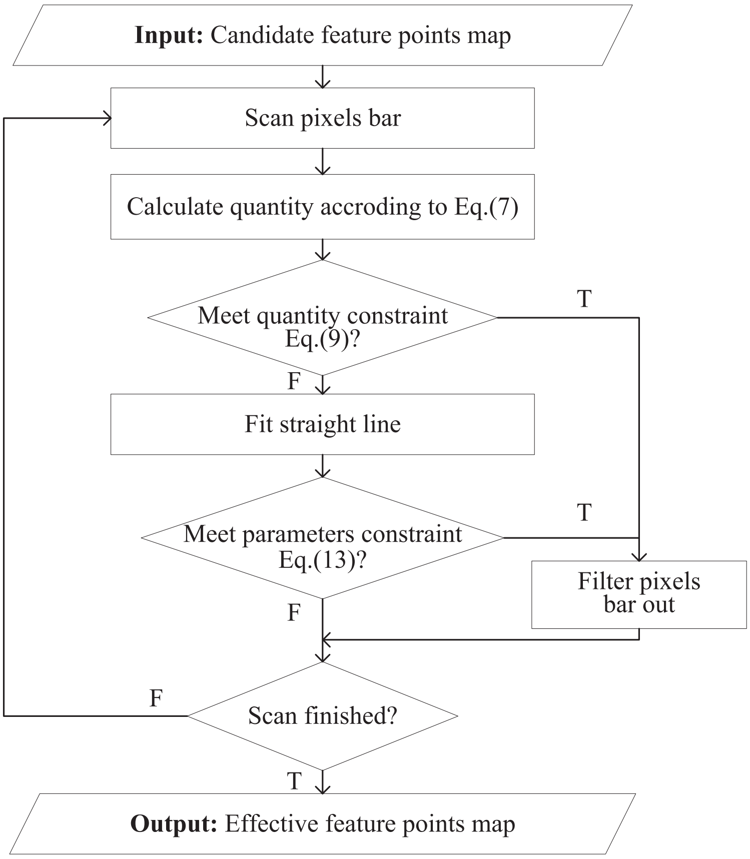

Input: Candidate feature points map
Scan pixels bar
Calculate quantity accroding to Eq.(7)
Meet quantity constraint Eq.(9)?
T
F
Fit straight line
Meet parameters constraint Eq.(13)?
T
Filter pixels bar out
F
F
Scan finished?
T
Output: Effective feature points map

FIGURE 5. Flow chart of filtering of pseudo feature points

Step 2: Check whether the scanned pixels bar satisfies the constraint of quantity. The criterion given in Eq.(9) is proposed to determine whether a pixels bar meets the condition of the feature of noises.

$$\zeta_1 = \begin{cases} 1, & P_{num} < \alpha_4 \\ 0, & otherwise \end{cases} \tag{9}$$

Step 3: Use the least squares method to fit a straight line, $y = kx + b$, representing a scanned pixels bar on the sample set Pos, where k, b are the slope and intercept values of the fitted straight lines, respectively.

Step 4: Check whether the slope k and intercept b of the scanned pixels bar satisfy the slope and intercept constraints. The pixels bar with morphological characteristics of noises in current image can be filtered out using the criterion given in Eq.(10) and Eq.(11).

$$\zeta_2 = \begin{cases} 1, & k \notin [-k_{max}, k_{max}] \ or \ k_L > 0 \ or \ k_R < 0 \\ 0, & otherwise \end{cases} \tag{10}$$

$$\zeta_3 = \begin{cases} 1, & |b - L_{mid}| > \alpha_5 \\ 0, & otherwise \end{cases} \tag{11}$$

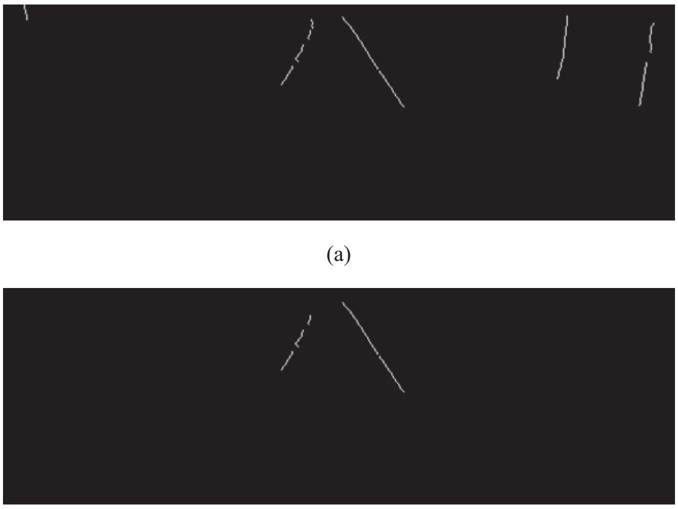
Step 5: Filter out the scanned pixels bar. The discriminant score ζ_{noise} obtained using the quantity constraint and the parameters constraint(including slope and intercept constraints) is then combined to identify whether the detected pixels bar is noise.

$$\zeta_{noise} = \begin{cases} 1, & if(\zeta_1 = 1 \; or \; \zeta_2 = 1 \; or \; \zeta_3 = 1) \\ 0, & otherwise \end{cases} \tag{12}$$

Step 6: If $\zeta_{noise} = 0$, we apply prediction constraint ζ_4 to further discriminate according to the lane with the property of consistently uniform along the time axis over a succession of frames.[1] Now, the discriminant score ζ_{noise} is as follows:

$$\zeta_{noise} = \zeta_{noise} \vee \zeta_4 \tag{13}$$

where $\zeta_4 = 1$ indicates that the pixels bar matches this constraint. The details of the prediction constraint ζ_4 are shown in section 3.4.



(a)

(b)

Figure 6. The result of the pseudo-feature points filtering. (a) results after quantity constraint, (b) results after parameters constraint.

Fig.6 shows effective feature points obtained after filtering out the pixels bars with ζ_{noise} is 1. In contrast to Fig.3, it can be seen that the pseudo-pixels bars, which containing small number of feature points and the parameters of its fitting line do not meet the requirements of constraints, are fitted, as shown in Fig.6(a) and Fig.6(b), respectively.

4. **Lane line extraction.** In this section, according to whether the set of effective feature points is empty, we use the following two different ways to obtain the lane line equations.

4.1. **Lane lines fitting.** If the set of effective feature points is not empty, we divide it into two parts according to the center line $y = L_{mid}$ of the ROI. The least squares method is a commonly used method for fitting straight lines. It can give a more accurate regression line in the sense of mean square error by minimizing the square of the error and finding the best function matching of the data [14]. And its computational cost is low, so that the real-time system can be improved. But the least squares method is more sensitive to noise, and very little noise can lead to errors in fitting results.

In the case of Fig.6(b), the least squares method yields the correct result. As shown in Fig.7, the lane lines on both sides are marked in blue and green, respectively. However, for Fig.8(a), the fitting result(Fig. 8(b)) is far from satisfactory. This is because all white pixel points in Fig.8(a) are involved in fitting line, even for noise points.

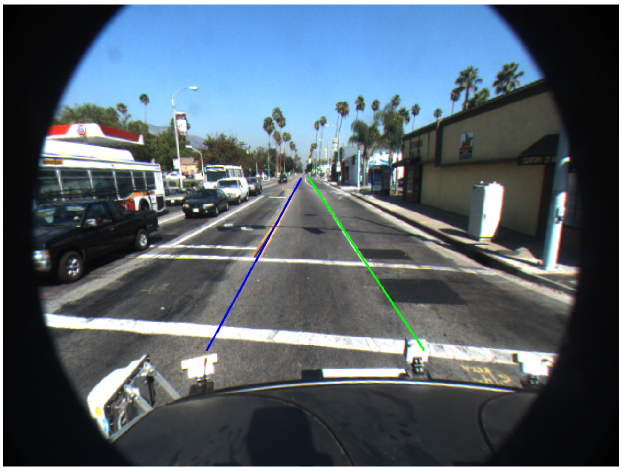


FIGURE 7. The result of lane fitting

To solve the problem of noise points, the RANSAC algorithm provides a good idea. It is a method of estimating the parameters of a mathematical model by iterative approach based on an observation data set containing outliers(noises) [15, 16]. The iteration times k of the algorithm can be inferred from the theoretical results, as shown in Eq.(14):

$$k = \frac{log(1-p)}{log(1-w^n)} \tag{14}$$

where p is the probability that the points randomly selected from the data set are inliers(valid feature points) during the iteration, w represents the probability of selecting an internal point from the data set, and n indicates the number of points that the estimated model needs to select.

The RANSAC can derive a model that is calculated only at the inliers and has a sufficiently high probability and good robustness. Fig.8(c) shows the results of the RANSAC for the effective feature points map in Fig.8(a), and the lane lines are correctly fitted. Therefore, the RANSAC algorithm is used to fit the lane lines in this paper.

4.2. **Lane lines prediction.** When the following phenomena occur, such as the temporary white light caused by the bumps of the vehicle or the shadows caused by the serious lack of light, etc. , they will lead to no feature points in the effective feature points set. The fitting algorithm can not be carried out. At this time, the prediction model is used to predict the lane lines.

Assuming that n images have been scanned, the detection result set R and the weight set W are computed according to Eq.(15) and Eq.(16), respectively.

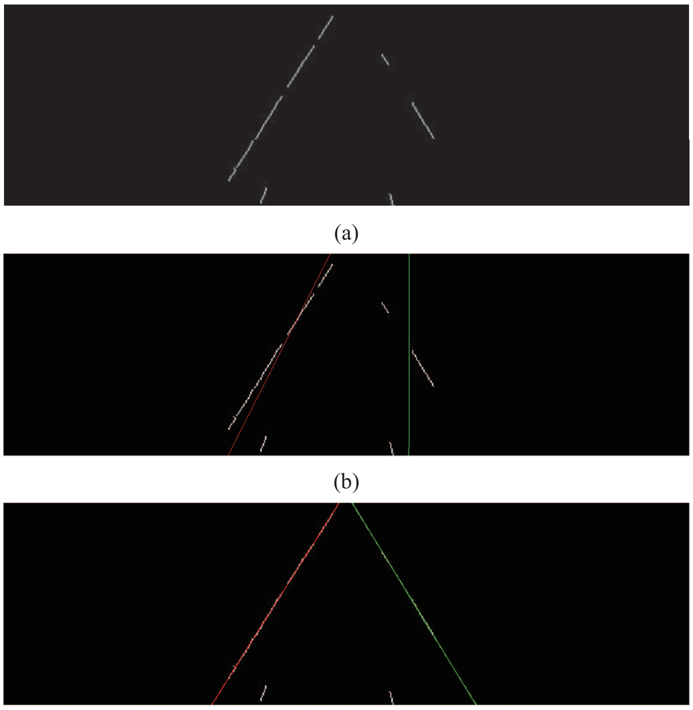
$$R = \{(k_i, b_i) | i = 1, 2, ..., n\} \tag{15}$$

$$W = \{w_i | w_i = \frac{i}{n+1}, i = 1, 2, ..., n\} \tag{16}$$

where k_i, b_i are the slope and intercept values detected or predicted of the i-th frame image, respectively. w_i is the prediction weight of the i-th frame image.

The lane line prediction model of the $(n+1)$-th frame is:

$$pred(n+1) = \{(k_{n+1}, b_{n+1}) | k_{n+1} = \sum_{i=1}^{n} w_i \cdot k_i / \frac{n}{2}, b_{n+1} = \sum_{i=1}^{n} w_i \cdot b_i / \frac{n}{2}, i = 1, 2, ..., n\} \tag{17}$$



(a)

(b)

(c)

Figure 8. Comparison of fitting results of RANSAC and least squares. (a) an effective feature points map with strong noises, (b) the least squares fitting results, (c) the RANSAC fitting results.

where k_{n+1} and b_{n+1} are the prediction values of the slope and intercept of the $(n+1)$-th frame image, respectively. Similar to the hidden layer update model of the recurrent neural network, the proposed prediction model memorizes and considers the information of the previous n frame images detected according to Eq.(17), and thereby can better predict the information of the next frame image. There is no doubt that memory will gradually forget over time, that is, the longer the image information generated, the less impact on the forecast [17].

In this paper, the predictive values are also used as a constraint to filter the pseudo-feature points simultaneously. The model for prediction constraint is as follows:

$$\zeta_4 = \begin{cases} 1, & |k - k_n| > \alpha_6 \ and \ |b - b_n| > \alpha_7 \\ 0, & otherwise \end{cases} \tag{18}$$

where α_6 and α_7 are two thresholds that limit the number of lane-like noises in the test results. When $\zeta_4 = 0$, it indicates that the detected pixels bar matches one lane, adding the pixels bar to the effective feature points set.

5. Experimental results and analysis.

The test environment of the proposed algorithm is as follows: OS is Windows 10, CPU is Intel(R) Celeron(R) G550 2.6GHz, RAM is 4GB. Tested images are from California Institute of Technology and our own image database. For the former, it has 1225 images, belonging to the urban daytime street scenario, and is a relatively complete lane detection database. Our own collection of photo databases holds three scenarios with 2340 images, including urban night street scenario, highway daytime scenario and highway night scenario. These pictures are not only of good quality, but also have a wealth of traffic information.

The environment analysis of the image databases is shown in Table 1. We counted the six categories of road conditions in the four test scenarios. Overall, there are three cases, lane discontinuous or partially missing, parking lines or passing vehicles and symbols or shadows on the road, accounting for a larger proportion. In urban day scenario(1225 frames), many lanes are discontinuous or partially missing, and in urban night scenario(702 frames), the frames which have parking lines or passing vehicles occupy the largest component. The proportion of bending lanes in highway day scenario(608 frames) is higher than in the other three scenarios. Highway night scenario(1030 frames) possesses a bit better traffic because it has the fewest disturbances of parking lines or passing vehicles. In the meantime, there are lots of parking lines or passing vehicles, symbols or shadows for urban day scenario.

Table 1. The environment analysis of the image database

Count Cate. Sce.	Little interfere	Discontinuous or missing	Parking lines or passing vehicles	Symbols or shadows	Bending	Other cases
Urban day	105	997	699	599	188	178
Urban night	27	305	668	346	60	94
Highway day	26	502	306	238	168	55
Highway night	174	659	96	342	126	34
Frames	332	2463	1769	1525	542	361

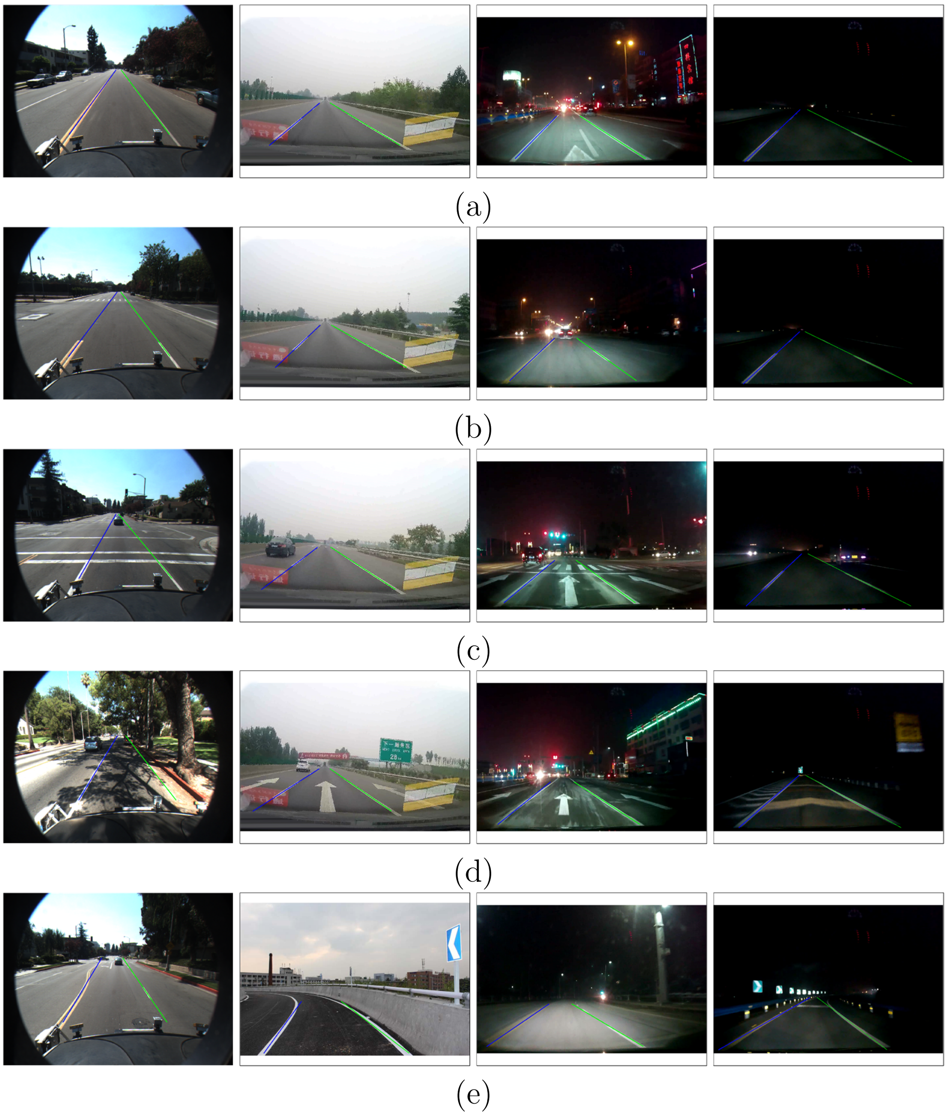
The following rules were used to classify the results into three categories: (a) correct detection, which more than 70% of the detected lane covered the lane in the scenarios; (b) incorrect detection, which the detected lane appeared in other areas; (c) missed, which lane line exists but failed to detect [18]. For these cases, such as little interference, parking lines or other vehicles, some texts and shadows interference on the road, and lane discontinuous or partially missing, etc., the proposed algorithm can quickly and effectively detect the lane lines, and the accuracy is more than 91%. Table 2 shows the results of the proposed method, which includes the number of frames of four test scenarios, the detected frames of proposed method, the correct detection rate, the error detection rate and the average detection rate.

Table 2. Statistics lane detection results based on ternary tree traversal

Scenarios	Urban day	Urban night	Highway day	Highway night	Total
Total frames	1225	702	608	1030	3565
Detected frames	1199	682	590	1012	3483
Correct rate(%)	92.93	87.26	90.53	95.40	91.53
Error rate(%)	7.07	12.74	9.47	4.60	8.47

From Table 2 we can see that the proposed algorithm can detect almost all the frames. Especially in highway night scenario, the algorithm has achieved relatively high accuracy because the lanes on this scenario are very prominent and noticeable after being illuminated by headlights. And the lowest detection rate is higher than 87%. Detection of a frame consumes about 116ms. Fig.9 shows some of the results of the lane detection. As can be clearly seen from Fig.9(a), the lane lines are correctly detected when there is a small amount of interference on the road surface. This is because lane information is more adequate and traffic is better.

When the lane is discontinuous or partially missing, the proposed algorithm can still detect the lane lines according to the small amount of effective information after filtering the noises, as shown in Fig.9(b).


(a)

(b)

(c)

(d)

(e)

FIGURE 9. Part of lane detection results. (a) little interference, (b) lane discontinuous or partially missing, (c) parking lines or other vehicles passing, (d) some symbols or shadows interference and (e) bending lane.

The results of the test with parking lines or other vehicles passing on the road are shown in Fig.9(c). In this situation, the characteristic points of parking lines or driving vehicles are discretized. The filtering algorithm based on ternary tree traversal is designed to filter out such pseudo-feature points, which makes the detection of such cases very simple and fast.

When the road has symbols and shadows interference, as shown in Fig.9(d), the shadows are filtered in the preprocessing stage, and most of the feature points of the symbols were discrete or the parameters of their approximate line do not meet the discriminant of the filtering algorithm. Using the proposed filtering algorithm, these pseudo-feature points can be effectively filtered.

Fig.9(e) shows the detection effect on the curved lanes. When the road is bent, the algorithm can dynamically adjust the parameters under the condition of prediction constraint to adapt to the change of lanes and obtain better test results.

In addition, we compared the other two lane detection methods[3], [6]. Yoo [3] detects each frame based on an improved gradient enhancement method for Canny edge detectors. The method is robust in a variety of environments because it maximizes the gradient between the road and the lane. Peng [6] detects lane lines based on inverse perspective mapping(IPM) and improved RANSAC method. The method is an all-lanes mode lane detection method, and we only evaluate its performance in the 2-lanes mode.

Table.3 shows the results of these methods under the four test scenarios. *Accuracy Rate* (*AR*) and $False\ Negative(FN)$ are used to evaluate the performance of these lane-line detection methods [19]. Peng's detection rate for urban night scenario and highway day scenario is relatively low since in the former scenario, most of the frames have passing vehicles, and the bending lanes in the latter scenario occupy a large part. In fact, the most basic reason of low detection rate of Peng's method is that it does not take into account changes in the environment and bending roads, and IPM-based lane-line detection is very prone to false detection when detecting forward vehicles on account of the forward vehicle is converted to a component like a lane. In our proposed method, the prediction constraints based on prediction model have improved the detection results of these conditions.

Because the gradient of the lanes and the lane-like noises was enhanced simultaneously, Yoo's method also has more false detection results. The method works not well in detecting the situation of parking lines and traveling vehicles, this is because the length of the parking line is often greater than the length of the lane line and it does not consider how to deal with strong transverse noises on the road. Moreover, Yoo utilizes Hough-based approach, which is time-consuming. Unlike Yoo's method, our approach makes horizontal noises discrete and easy to filter them out. Meanwhile, the simple math calculations and discriminant we used improves the performance of the algorithm.

Table 3. The performances of different detection methods under four test scenarios

Scenarios	Yoo's method		Peng's method		Proposed method	
	AR(%)	FN(%)	AR(%)	FN(%)	AR(%)	FN(%)
Urban day	88.98	5.03	82.86	3.00	92.93	3.05
Urban night	80.38	4.31	73.94	3.40	87.26	2.26
Highway day	82.34	3.02	75.20	5.66	90.53	4.30
Highway night	90.90	1.65	86.29	2.01	95.40	1.45
Total	85.65	3.50	79.57	3.52	91.53	2.77

The proposed method has achieved good performance on the whole owing to it uses a dynamic noises filtering method, which taking into account the change information in the previous lanes. On the curved road, it has a better test results. Compared with the Hough-based method, there is a low false positive rate and time consumption. Indeed, the proposed prediction algorithm is similar to the update algorithm of the hidden layers in recurrent neural network, and predicts the lane lines of the current frame according the trend of the past to the present of the lanes. In urban night scenario, the test results of the proposed method are also relatively low because of the complicated road conditions and heavy traffic at night. But compared with the other two methods, the detection performance is improved because the proposed method considers the temporal consistency [1] of the continuous frames and uses the ternary tree traversal based noises filtering algorithm, which utilizes four constraints to determine the presence of noises and can effectively filter them out.

Of course, the proposed method can not work well when the road has a strong shadow interference. Fig.10 shows the false detection and missed detection. This is because most of the lane lines under strong shadows are filtered as background pixels during the preprocessing process, which leads to the detection error. Moreover, when the lanes are seriously missing or no lanes exist, the detection will be wrong although the prediction algorithm is used, this is due to the accumulation of errors in the previous frames detection. The processing of these cases is our next work.

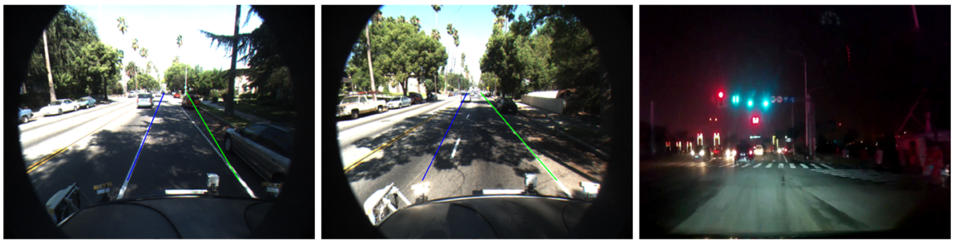


FIGURE 10. False detection and missed detection

6. **Conclusion.** This paper presents a lane detection algorithm combining the lane feature points information and the ternary tree traversal idea. The algorithm firstly extracts the candidate feature points based on the left and right edges of the lane, and then uses the proposed filtering algorithm independent of the parameter space to filter the pseudo feature points. A lane prediction is used to predict lane lines for special traffic conditions. Experiments show that the detection rate is more than 91% when detecting the part of lane missing, vehicle passing and symbols interference on the road, etc., and the average detection time of single image is 116ms. In future research we will further study the extraction of feature points in strong shadows and how to improve the detection rate of lane marks of the current image by using the detection results of the detected images.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (61572173, 61472119, 61401150 and 61472373), Henan Provincial Department of Science and Technology Research Project (172102110032), the Key project of Henan Provincial Education Department (17A210014), Henan Science and Technology Innovation Outstanding Youth Program (184100510009) and the Fundamental Research Funds for the Universities of Henan Province (NSFRF1604).

REFERENCES

[1] S. Jung, J. Youn, S. Sull, Efficient lane detection based on spatiotemporal images, *IEEE Transactions on Intelligent Transportation Systems*, vol.17, no.1, pp.289-295, 2016

[2] A. Mammeri, A. Boukerche, Z. Tang, A real-time lane marking localization, tracking and communication system, *Computer Communications*, vol.73, pp.132-143, 2016.

[3] H. Yoo, U. Yang, K. Sohn, Gradient-Enhancing Conversion for Illumination-Robust Lane Detection, *IEEE Transactions on Intelligent Transportation Systems*, vol.14, no.3, pp.1083-1094, 2013.

[4] J. F. Wang, F. D. Gu, C. Zhang, and G. Z. Zhang, Lane boundary detection based on parabola model, *IEEE International Conference on Information and Automation*, Harbin, China, pp.1729-1734, 2010.

[5] A. Borkar, M. Hayes, M. T. Smith, A Novel Lane Detection System With Efficient Ground Truth Generation, *IEEE Transactions on Intelligent Transportation Systems*, vol.13, no.1, pp.365-374, 2012.

[6] H. Peng, J. S. Xiao, S. M. Shen, B. J. Li, and X. Cheng, A fast algorithm based on RANSAC for vision lane detection, *Journal of Shanghai Jiaotong University*, vol.48, no.12, pp.1721-1726, 2014.

[7] J. S. Xiao, X. Cheng, B. J. Li, W. Gao, and H. Peng, Lane detection algorithm based on beamlet transformation and K-means clustering, *Journal of Sichuan University: Engineering Science Edition*, vol.47, no.4, pp.98-103, 2015.

[8] S. Yenikaya, G. Yenikaya, E. Düven, Keeping the vehicle on the road: A survey on on-road lane detection systems, *ACM Computing Surveys*, vol.46, no.1, pp.125-134, 2013.

[9] J. F. Wang, Y. Wu, Z. H. Liang, and Y. J. Xi, Lane detection based on random hough transform on region of interesting, *IEEE International Conference on Information and Automation*, Harbin, China, pp.1735-1740, 2010.

[10] G. Q. Jiang, Z. H. Wang, C. J. Zhao, An Algorithm of Detecting Crop Rows Based on Known-points, *Journal of Basic Science and Engineering*, vol.21, no.5, pp.983-990, 2013.

[11] Q. Q. Li, L. Chen, M. Li, S. L. Shaw, and A. Nüchter, A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios, *IEEE Transactions on Vehicular Technology*, vol.63, no.2, pp.540-555, 2014.

[12] X. Y. Li, X. Z. Fang, C. Wang, and W. Zhang, Lane detection and tracking using a parallel-snake approach, *Journal of Intelligent and Robotic Systems*, vol.77, no.3-4, pp.597-609, 2015.

[13] I. S. Gruzman, Threshold binarization of images based on the skewness and kurtosis of truncated distributions, *Optoelectronics, Instrumentation and Data Processing*, vol.49, no.3, pp.215-220, 2013.

[14] M. Montalvo, G. Pajares, J. M. Guerrero, J. Romeo, M. Guijarro, A. Ribeiro, J. J. Ruz, J. M. Cruz, Automatic detection of crop rows in maize fields with high weeds pressure, *Expert Systems with Applications*, vol.39, no.15, pp.11889-11897, 2012.

[15] Y. G. Liu, B. L. Sun, Y. T. Shi, Z. X. Huang, and C. H. Zhao, Stereo Image Rectification Suitable for 3D Reconstruction, *Journal of Sichuan University: Engineering Science Edition*, vol.45, no.3, pp.79-84, 2013.

[16] Q. Miao, Z. L. Fu, X. H. Zhao, and K. J. Xu, Parallel Improved RANSAC Based on CUDA, *Journal of Sichuan University: Engineering Science Edition*, vol.42, no.4, pp.111-116, 2010.

[17] J. Li, X. Mei, D. Prokhorov, and D. C. Tao, Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene, *IEEE Transactions on Neural Networks and Learning Systems*, vol.28, no.3, pp.690-703, 2016.

[18] J. Son, H. Yoo, S. Kim, and K. Sohn, Real-time illumination invariant lane detection for lane departure warning system, *Expert Systems with Applications*, vol.42, no.4, pp.1816-1824, 2015.

[19] J. W. Niu, J. Lu, M. L. Xu, P. Lv, X. K. Zhao, Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting, *Pattern Recognition*, vol.59, pp.225-233, 2016.