

Multi-lane Detection Based on RMFP For Self-Driving in urban traffic scenes

Chao LI^b, Hong-zhe LIU^a, Yong-Rong ZHENG, Han-yu XUAN

Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing
^aliuhongzhe@buu.edu.cn, ^bgeoff@buu.edu.cn

Keywords: Multi-lane detection, Autonomous vehicle, RMFP, IPM, Kalman Filter

Abstract. The lane detection is important for the autonomous vehicle vision navigation used in the intelligent transportation system (ITS). Several approaches for lane detection were suggested in the past. However, there is still one issue about robustness. This paper presents a robust and real-time multi-lane detection method based on Road Marking Feature Points (RMFP) for autonomous vehicle navigation in urban environment. The key idea is to apply methods from extracting RMFP and the target tracking domain to identify lanes information in this article. Then we extract the RMFP from the gray-scale image and the IPM image. Besides we also use the lane line color and structure features to sift RMFP that meets lane line. At last, we adopt the clustering method to generate lane lines, and we track these lines by frame association and Kalman Filter. The experimental results show that our proposed method is robust and real-time detect the lane line of various kinds of complicated road. And based on the lane line visual navigation of unmanned experiment validate the reliability of our method.

Introduction

Recent years, advances in self-driving technologies have captured much attention and interest from researchers, the media and the general public. With the Google's self-driving car driving officially for testing on the California highway in 2015, other giant auto makers or corporations (such as Mercedes, Tesla, Volvo, BMW etc.) have been generally acknowledged that it is just a matter of time before self-driving cars step into people's life and have great influence on people's exist state. Therefore, a real-time and robust lane detection method is very important for autonomous vehicle vision navigation. There are many existing lane line method such as IPM-based, Hough transform-based, edge-based and threshold-based methods. Each method has its own advantages and disadvantages. For IPM-based in [1]-[14], the difficulty involved the calibration of the camera. If the camera position is changed will affect the lane line detection result. But the IPM image has the very good parallelism. In [5], Wang used Inverse Perspective Mapping to transform binary image space to top view space and using k-means clustering algorithm to analysis linear discriminant for reducing interference affect. For edge-based methods[6],[17],[21], edge detection preserves important structural properties and significantly reduces the amount of data, but the difficulty involved eliminating noise edges from complex scenarios. The lane-mark edges might merge with the edges of shadows or dirt or cracks. For threshold-based approaches in [5]-[8],[15],[18]-[20] are sensitive to changes in illumination.

Most of the existing lane detection methods are designed for Advanced driver assistance system (ADAS), for example lane departure warning system(LDWS). Most of the existing lane detection methods in [5]-[10],[15]-[24] only detect the current lane, however, multi-lane detection is very necessary for autonomous vehicle vision navigation when vehicle is changing lane. Several methods of multi-lane detection are introduced in [4],[11],[26]. Li [12] and Fu [13] were applied to omni-directional and panoramic camera, respectively. Their method than the average speed of monocular camera slowly.

In this paper, we proposed a robust and real-time method for multi-lane detection based on Road Marking Feature Points(RMFP) for autonomous vehicle navigation in urban environment and introduced an approach of RMFP extraction. Our proposed method has simple steps. Only interested in locale image preprocessing and image grayscale, no use of the methods of dealing with the time consuming and not robust such as edge detection, threshold processing and Hough transform. From

inverse perspective projection of gray image get a bird's eye view of the image, then from the gray image and the bird's eye view extract RMFP. At last, we adopt the clustering method based on lines to generate lane line, meanwhile, tracking by frame association and Kalman Filter. In a number of city road experiments prove our multi-lane line detection method is more than the existing lane line detection method is more robust and real-time. Our method was successfully tested in the China Future Challenge 2014 and 2015 (CFC 2014, CFC 2015), which is an intelligent-vehicle competition.

The organization of this paper is as follows. In Section II, we describe the details of the proposed multi-lane detection method. Section III describes some experimental results under various urban conditions to confirm the excellence of the proposed method. Finally, Section IV discusses our conclusion and future works.

Image preprocessing

In this paper, several preprocess steps are required.

First, we design a customized mechanism to determine image quality to prevent over or under exposure. This is helpful especially to obtain adjusted camera exposure time when the vehicle moves from a shaded area to an unshaded or the other way round.

Lane line from the color, geometric edge and other road areas there is a significant difference, lane line color mainly is white or yellow. The most influential factors are R and G in the RGB color space. In this paper, the more suitable gray-scale conversion algorithm, expressed as:

$$G_{gray} = R * 0.45 + G * 0.45 + B * 0.1$$

One crucial step, gray-scale changed on a small scale own the same grey value in a image. Therefore, we optimize the result from median filtering within domain window, and calculate the mid-value only in an isochronous relation between a new generated window and out of bounds window pixel gray value. Compared with the original median filtering method, this method reduces the noise interference, and can improve the efficiency of the time. In this paper, the image is filtered using a 3 * 3 square neighborhood window.

Inverse Perspective Mapping

Inverse Perspective Mapping is a process that converts perspective images to inverse perspective images by means of transformation matrix . Being able to get the solution formula of the H in IPM is that :

$$H = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}^{-1} \quad (1)$$

We can know that there are eight unknown numbers in the H matrix according to the formula(1),we need eight coordinate points to solve the H matrix ,including four world coordinate points of $P_1(X_1, Y_1)$, $P_2(X_2, Y_2)$, $P_3(X_3, Y_3)$, $P_4(X_4, Y_4)$ and four image coordinate points of $p_1(x_1, y_1)$, $p_2(x_2, y_2)$, $p_3(x_3, y_3)$, $p_4(x_4, y_4)$. Coordinate points in two groups are opposite to each other that is to say there is a mapping relation between point p1 in the image and point P1 in the world coordinates. To get those coordinates ,we can put four obvious markers on the pavement, such as conical barrel, as shown in figure1 ,rectangular four peaks happen to be the four points we need.

Image coordinate points :image coordinates that rectangular four peaks of $P_1 P_2 P_3 P_4$ in perspective image obtained by way of clicking the mouse manually, which is shown in figure 2 .There are certain operating errors among image coordinates which gained by way of clicking the mouse manually, but this type of errors can be accepted by practical application.

World coordinate points :It need to build the world coordinate system which using pavement as plane and vehicle location as reference before calculating the world coordinate points ,which is shown in figure 3. $View_W$ and $View_H$ in figure 5, respectively, on behalf of the width and height of the

bird's-eye view. R_W and R_H , respectively, represent the actual width and height of rectangle which is made up of point P_1 , point P_2 , point P_3 and point P_4 , while Dis represents the actual distance between the bottom of the rectangle and vehicle, and their unit all are centimeter .

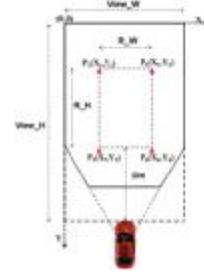


Fig.1 Obtaining image coordinates through clicking mouse. Fig.2 Coordinates of rectangle's four peaks in the world coordinate system.

The homography matrix H transforms the perspective image into the bird-eye view image, so those world coordinate points should be transformed into the coordinate points of the bird-eye view, while the bird-eye view is the image expression of the plane of the world, there is the equal proportional relationship between it and the real plane of the world, we suppose the proportion coefficient is k (cm/pix), so the width of every pixel in the bird-eye view represents k -cm practical pavement. Then we can get to know the numeration of the world coordinate point according to figure 5, and the numeration as follow:

$$\begin{bmatrix} X_1 & Y_1 \\ X_2 & Y_2 \\ X_3 & Y_3 \\ X_4 & Y_4 \end{bmatrix} = \begin{bmatrix} (View_W - R_W)/2k & (View_H - Dis - R_H)/k \\ (View_W + R_W)/2k & (View_H - Dis - R_H)/k \\ (View_W - R_W)/2k & (View_H - Dis)/k \\ (View_W + R_W)/2k & (View_H - Dis)/k \end{bmatrix} \quad (2)$$

In those data, the value of R_W , R_H and Dis is get through measuring, while the value of $View_W$, $View_H$ and k can be set according to the visual needs. In the textual experimentation process, we adopt the parameter setting that $View_W = 1500$, $View_H = 2500$ and $k = 5$, so the image size of the bird-eye view is 300×500 .

RMFP Extraction

Due to road identification and the road has obvious contrast, which makes the pavement logo has a unique feature, we will be this kind of features is called the pavement marking, according to the pavement marking points feature extraction is pavement marking feature points.

We analysis the road image in a row to the gray value of intensity distribution. We can find from the figure 3 and the figure 4 have jumped points in the edge of the pavement marking points which distributed on both sides. We call up along points to the left of points, and call the right side points as falling edge points, and these points are collectively referred to as pavement marking feature points.

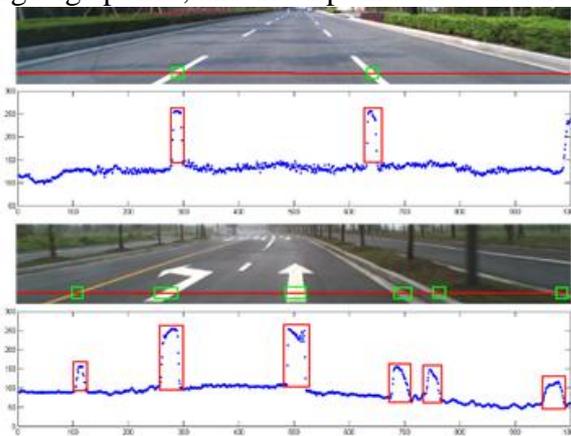


Fig.3 PM of a line intensity of gray value distribution

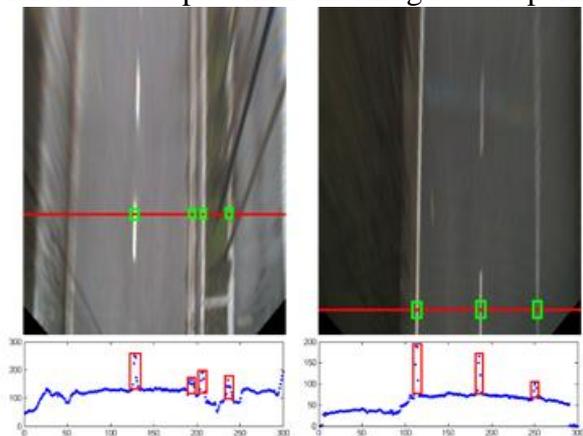


Fig.4 IPM of a line intensity of gray value distribution

Because the pavement marking has obvious rising and falling, so it needs to extract one point individually on rising edge and falling edge to make up a pair of points when we extract pavement marking feature points. The pavement marking feature points belong to the bigger jumped points long with adjacent pixel gray value's D-value. Thus, we adopt local gradient method to extract the road marking feature points.

Firstly, we calculates the average gray value $avgV$ of the gray image's local road to access the pavement's gray intensity, whose location near the below in general. When the camera is fixed, we can specified for a region as the local surface area. We adopt uniform sampling method, namely every area extract a pixel gray value in the column k_i and row k_j , and the calculation formula $avgV$ is as follows:

$$avgV = \frac{1}{N} \sum_{i=I_{start}/k_i}^{i=I_{end}/k_i} \sum_{j=J_{start}/k_j}^{j=J_{end}/k_j} f(i \cdot k_i, j \cdot k_j) \quad (3)$$

Among them, $f(i \cdot k_i, j \cdot k_j)$ on behalf of the gray levels of pixels to $(i \cdot k_i, j \cdot k_j)$, N for the number of sampling pixels, I_{start} , I_{end} and J_{start} , J_{end} represent the start and pixels' column and row, and k_i and k_j respect increment of the column and the row.

Secondly, we calculate pixel point (x, y) horizontal proximity pixel gray average value avg and access the pixel gray value's intensity level. When the gray value is too high or too low near the pixel, the gradient between the pavement marketing and the road's gray will become small. The average of gray's calculation formula is as follows:

$$avg_{(x,y)} = \frac{1}{t} \sum_{i=-t/2}^{i=t/2} f(i+x, y) \quad (4)$$

When $t=5$, it can achieve the desired effect.

Meanwhile gradient threshold value formula of ΔT is as follows:

$$\Delta T = \begin{cases} avg / t_h, & avg > 200 \\ avg / t_m, & avgV < avg \leq 200 \\ avg / t_l, & 30 < avg \leq avgV \end{cases} \quad (5)$$

Among them, t_h , t_m , t_l are the tune parameter of high bright area、 medium bright area and low bright area. The parameter that we used in the experiment are : $t_h=12$, $t_m=6$, $t_l=10$. As shown in figure7, A, B, C are high bright area、 medium area and low bright area individually and we can see that the gradient range of high bright area and gray area are small, but medium bright area's gradient is bigger than them.

Calculate the pixel point's gray value's gradient Δv before and after pixel point (x, y) , and the calculation formula is as follows:

$$\Delta v = f(x, y+1) - f(x, y-1) \quad (6)$$

Among them, $f(x, y+1)$ and $f(x, y-1)$ are pixel point (x, y) horizontal direction's the last one pixel point $(x, y+1)$ and the previous one pixel point's $(x, y-1)$ gray value, when gradient $\Delta v > 0$, it's rising edge's points and if $\Delta v < 0$, it's falling edge's points.

When we are extracting REFP, we need extract rising edge's points firstly, and then extract falling edge's points, and these can be composed of a feature points pairs. We assume $P \uparrow (x, y)$ and

$P \downarrow(x, y)$ that impress the nature of the point, when $P \uparrow(x, y) = 1$ impress pixel point (x, y) is rising edge point, otherwise it's not rising edge point. And in this way, when $P \downarrow(x, y) = 2$ express pixel point (x, y) is falling edge point. The calculate formulation of $P \uparrow(x, y)$ and $P \downarrow(x, y)$ is as follows:

$$P \uparrow(x, y) = \begin{cases} 1 & \Delta v \geq \Delta T, f(x, y) > 30 \\ 0 & \text{else} \end{cases} \quad (7)$$

$$P \downarrow(x, y) = \begin{cases} 2 & \Delta v \leq -\Delta T, f(x, y) > 30 \\ 0 & \text{else} \end{cases} \quad (8)$$

According (7) and (8), we can extract rising edge point and falling edge point out from a line's pixel in the image. From the figure 5, red represents rising edge point and blue represents falling edge point. We see that on the left of the lane line are rising edge points and the right's are falling edge points.

We require road marking feature points are the points pairs by composed of adjacent rising edge points and falling edge points. As shown in figure 6, we attribute all the pixel points' assignment in a line, which rising edge points are 1 and falling edge points are 2 and other points are 0. These are impressed by red, blue and green as shown in figure, then put 0 point out according to the original order, again to find two adjacent values are 1 and 2 pixels, there is a note that must be in front of a value of 1 point and the value of 2 is right.

According to the above methods, we can extract road marking feature points out, the results is as shown in the figure 7.

Above them are aiming at extracting road marking feature points in a line pixel on the images. But compare to extract feature in the whole image, which is every line's above ways to extract in the image. however, when the lane line detection does not need so many feature points, in order to improve the extraction rate of feature points tend to adopt a few line extraction method, to extract the feature points in the image at the same time you just need to on the road area is extracted. Perspective images to extract the road marking the result of the feature points as shown in figure 8, and the result of inverse perspective image extracted as shown in figure 8.

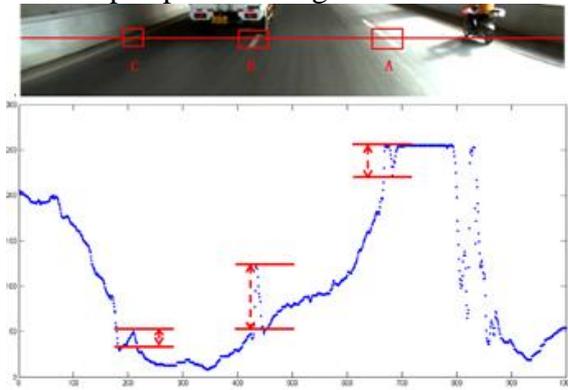


Fig.5 High bright area、 medium bright area and low bright area's gradient condition

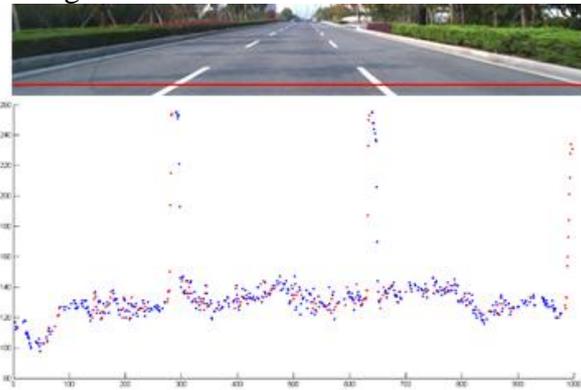


Fig.6 rising edge points and falling edge points' extracting result

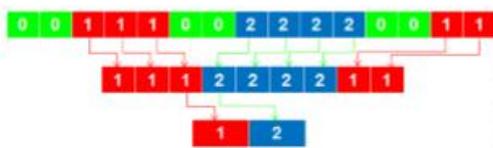


Fig.7 RMFP points for the extraction of the diagram

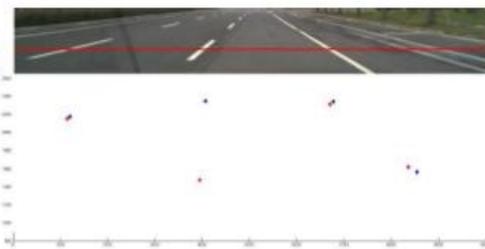


Fig.8 a line of pixels in pavement marking feature points extraction results



Fig.9 the results of extracting RMFP from PM

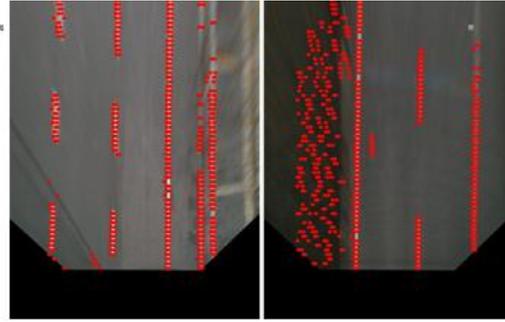


Fig.10 the result of extracting RMFP from IPM

In order to make full use of the respective advantage of perspective images and inverse images, when we are extracting RMFP, we will be extracted from two kinds of image points together to enhance RMFP extraction effect. Before integration needs to be extracted from the perspective image RMFP in inverse perspective projection image. Figure 18 is a process of fusion feature points.

From figure 9 and figure 10 can extract feature points were found to contain a lot of noise, it is because the above result is only based on the edge of the pavement marking points of feature extraction. In order to extract the results better, we also need to according to the width of the road marking feature extraction of feature points for each logo, which like the width of the lane line is 10 cm to 15 cm (in China) and a zebra crossing the line width is usually 40 cm or 45 cm. Assume $width_lanes$ represents lane line's width in true, and k_IPM express the true long of every pixel in inverse perspective in the image and in this article , k_IPM is $5cm/\pi x$, and lane line's width w_lane calculation formula is as follows:

$$w_lane = width_lanes / k_IPM \quad (17)$$

The lane line width in the inverse perspective images, we can according to the width of the lane line features to constraint RMFP to filter out noise points do not conform to the lane line width, the specific constraints formula is as follows:

$$RMFP(P_i, P_r) = \begin{cases} RMFP & , (W_{P_i P_r} - w_lane) \leq \Delta w \\ noise & , else \end{cases} \quad (18)$$

Among them, is the admitted deviation range, we set up the deviation's admitted range $\Delta w = 0.5$, $W_{P_i P_r}$ is feature points pairs $P_i(x_i, y_i)$ and $P_r(x_r, y_r)$ lateral separation, and the calculation formula is as follows:

$$W_{P_i P_r} = |x_r - x_i| \quad (19)$$

After though the formulation (18)'s width feature restrict filtering, which could filter a part noise points, and the result is as shown in the image 11.

From the figure 13 shows, through the lane line width after feature constraint filter can filter out most of the noise, but there are still a lot of noise, so the need for further filtering. RMFP extraction process in the use of the above the edge of the lane line, width, but the color of the lane line still has an important feature, so the next filter for color features RMFP constraints.

Before the color feature constraint filter need to use on the extraction of feature points on $\langle P_i(x_i, y_i) | P_r(x_r, y_r) \rangle$ computing center $P_c(x, y)$, calculation method as shown in the following formula:

$$P_c(x, y) = \begin{cases} x = (x_i + x_r) / 2 + \Delta x \\ y = y_i \parallel y = y_r \end{cases} \quad (20)$$

Among them, Δx is bias, which is for guaranteeing $P_c(x, y)$ on the lane line, in this article $\Delta x = 1.0$. Because points pair $\langle P_i(x_i, y_i) | P_r(x_r, y_r) \rangle$ on the same line pixel, just $y_i = y_r$, so $y = y_i$ or $y = y_r$ is the same.

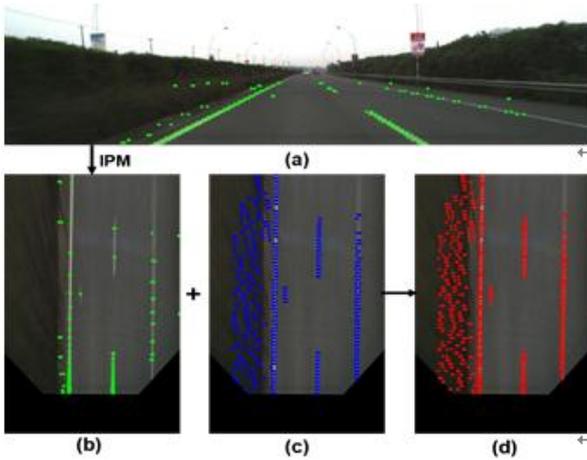


Fig.11 is a process of fusion feature points. Among them, (a) is the RMFP extracted from perspective images, (b) is the perspective image to extract the RMF mapped to inverse perspective in the image, (c) is extracting from inverse perspective in the image. (d) is the result of the compromises of (b) and (c)'s feature points.

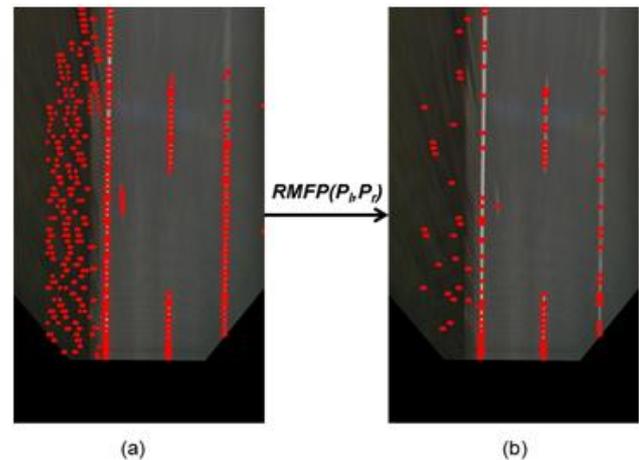


Fig.12 the result of the width feature restrict filtering (a) after the combination of feature's RMFP (b) filtered RMFP

In order to enhance the filtering effect, we adopt double-layer filtering, namely on inverse perspective image and image at the same time. At first, we determine whether the point $P_c(x, y)$ is satisfied with the lane's color feature in inverse perspective image. We assume $f(x, y)$ represents point's $P_c(x, y)$ gray value in inverse perspective and $g(x, y)$ represents point's $P_c(x, y)$ gray value reflected into the perspective image. So we have the followed color restriction formulation:

$$RMFP(P_c) = \begin{cases} RMFP & , f(x, y) > \overline{grayV_f} + \Delta v_f \\ & \& g(x, y) > \overline{grayV_g} + \Delta v_g \\ noise & else \end{cases} \quad (21)$$

Among them, $\overline{grayV_f}$ and $\overline{grayV_g}$ express average gray value respectively in road area of invest perspective image and perspective image, and the specific calculation ways could refer to the formulation (11). But Δv_f and Δv_g express the regulated quantity respectively in invest perspective image f and perspective image g , which are used to regulate and charge filtering and it's good. The figure 13 shows the progress and result of color feature filtering. The above is the all principles and processes of extracting lane line RMFP. Due to the above ways, we can get the better lane line RMFP, but to zebra crossing, guide arrow and these pavement markings also could by this way to come true, and only need to revise the parameters of the width feature filtering.

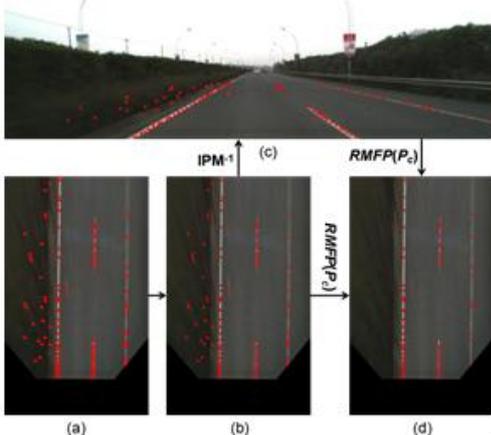


Fig.13 the progress and result of color feature filtering (a) the result of width feature filtering, (b) the result of calculate center point by the feature points showed in (a), (c) the result of though perspective reflected on perspective image, (d) the result of though two colors' feature filtering by (b) and (c).

Clustering Based on Lines

The basic idea of Hough transform is to use the duality of point and line. The given straight line in the original image space is transformed into a point in the Hough space by the curve expression. In this way, the detection problem of a given line in the original image is transformed into the problem of finding the peak value in Hough space. Compared to the classical Hough transform, the probability based Hough transform is able to select pixels randomly. When the straight line meeting the condition, the feature points on the line are removed to reduce the amount of computation of the algorithm on some pixels. Constrained by the algorithm itself, however, Hough transform has some limitations. When the linear detection of Hough transform is performed on the features containing noise. Because of the resolution is too small, the number of units in the coarse resolution will be dispersed to a number of smaller discrete parameter units. The accumulator can not get accumulated sufficient votes caused by failure detection. Therefore, the Hough transform is affected by the noise, so it can only get limited detection accuracy.

The least squares method is one of the commonly used linear regression method, can obtain a given data set is absolutely accurate linear in the mean square error, and higher detection accuracy than HT. However, LZ is sensitive to the outlier that deviates significantly from the regression line, and a small amount of strong noise significantly changes the regression line obtained by the least squares.

In the actual complex road environment, there are inevitably introduction noise in the characteristic stage of extracting lane line. Considering the advantages and disadvantages of Hough transform and least square method, this paper proposes a linear detection method combining those.

Firstly, the approximate area of straight lines is determined by Hough transform, and then the precise linear parameters are determined by using the improved least squares method for the feature points in each region.

Lane Lines Tracking

In the actual acquisition system and most of the intelligent vehicle systems, vehicle cameras directly obtain the video stream information, video streams in the adjacent two images are often between the great redundancy. Vehicle movement in time and space are continuous, because the car camera sampling frequency, in the image frame sampling period, the vehicle is only a very short distance forward, the road changes in the scene is very small, The performance of the front and rear lane change between the lane position is slow, so the previous frame image for the next frame image provides a very strong lane line position information. In order to improve the stability and accuracy of the lane recognition algorithm, this paper introduces the inter-frame association constraint.

Suppose there are $m \in N$ lane boundaries L estimated from the edge features at the current frame and $n \in N$ existing lane boundaries E and their corresponding Kalman filters from the previous K frame, boundary in L to determine the affiliation between them

$$L = \{L_1, L_2, \dots, L_m\}, E = \{E_1, E_2, \dots, E_n\}, K = \{K_1, K_2, \dots, K_n\}$$

We first build a $C = m \times n$ where the element in the i -th row and j -th column represents the distance Δd_{ij} between L_i and E_j .

$$\Delta d_{ij} = [x_i^{L_A} - x_j^{E_A}, C \cdot (x_i^{L_B} - x_j^{E_B})]^T \in R^2,$$

which A, B , respectively, represents the L_i, E_j , the upper and lower endpoint. Next, in the $R = \{1, 2, \dots, i\}$ row, the number e of Δd_{ij} in each i -th row is counted in $\Delta d_{ij} < T$.

If $e < 1$, the current lane line information does not satisfy the associated previous frame as the new lane line edge and delete the i in the R . If $e = 1$, the current L_i and previous frame lane edges E_j can be considered to be the same in continuous space, then set $V_i = \Phi$ and delete the i in the R .

$$\text{If } e > 1, V_i = \{\Delta d_{i1}, \dots, \Delta d_{ij}\}, \Delta d_{ij} = \begin{cases} 0, & \Delta d_{ij} > T \\ \Delta d_{ij}, & \text{other} \end{cases}$$

Statistic the number V_j of Δd_{ij} in $\{V_i\}$ where $\Delta d_{ij} \neq 0$, and $(\Delta d_{ij})_{\min} = \min\{V_j\} (V_j \neq 0)$

$\exists (\Delta d_{ij})_{\min} > 0$, it is obtained that the current frame L_i and the previous frame E_j are the same in the continuous space, set $V_j = \Phi$ and delete the i in the R .

It is over if $R = \Phi$.

Through the above method, we can relate to the current frame and the previous frame lane line information. But we want more stable information, such as the current frame and the first three frames of the association, this method is not applicable. Kalman is an excellent solution for this problem.

Algorithm Summary

For better illustration, the algorithm is summarized in the following flow diagram (Figure.14), where the italics indicate the output of the previous process.

The algorithm starts from Monocular camera and ends with vehicle lane information. The lane line position prediction module makes use of the current estimation results to predict the lane line position in the consequent image to improve the image processing consistency. One of the most detailed is the RBPF algorithm, yet is the innovation of this paper.

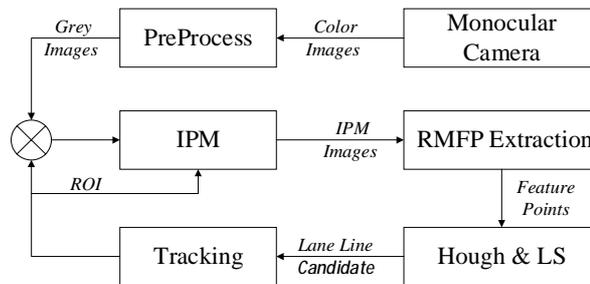


Fig.14 Lane line detection algorithm flow diagram.

EXPERIMENTAL RESULTS

The proposed algorithm is implemented in Opencv3.0 running on a Windows PC with Intel i5 CPU. In our experiments, the camera is installed in the car directly under the front windshield center position, from the ground distance of 1.5 m, and the camera's optical axis parallel to the plane of the vehicle chassis, facing the front of the vehicle. The acquisition cycle is 25ms.

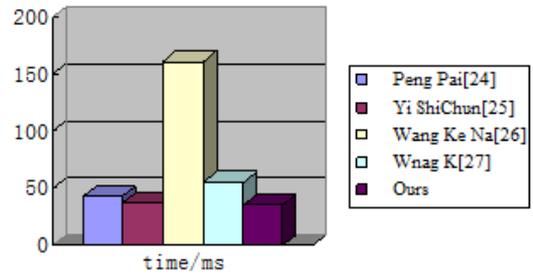
Table 1 lists the experimental results in a variety of complex driving environment data, S is the total frames, MPR is the missed positive rate, FPR is the false positive rate, and TPR is true positive

rate . Meanwhile, we tested the real-time algorithm, the algorithm average processing time per frame of about 35ms, much higher than in Table 3 other literature algorithms time.

Table1 Results in variety driving environment data

	Factors	S	MPR	FPR	TPR
1	normal	15367	0.6%	1.1%	98.3%
2	shadow	9925	0.5%	1.7%	97.8%
3	damaged	11343	1.4%	0.5%	98.1%
4	poor-exposure	6243	1.8%	1.7%	96.5%
5	over-exposure	15343	1.7%	0.8%	97.5%
6	fog	9561	3.1%	1.0%	95.9%
7	rain	13290	0.6%	0.9%	98.5%
8	snow	8424	1.3%	3.5%	95.2%
9	dusk	7898	1.2%	0.6%	98.2%

Fig.15 Runtime of different algorithm



Meanwhile, we tested the real-time algorithm, the algorithm average processing time per frame of about 35ms, much higher than in Figure 16 other literature algorithms time. Figure 22 shows the lane line detection results for different road conditions.

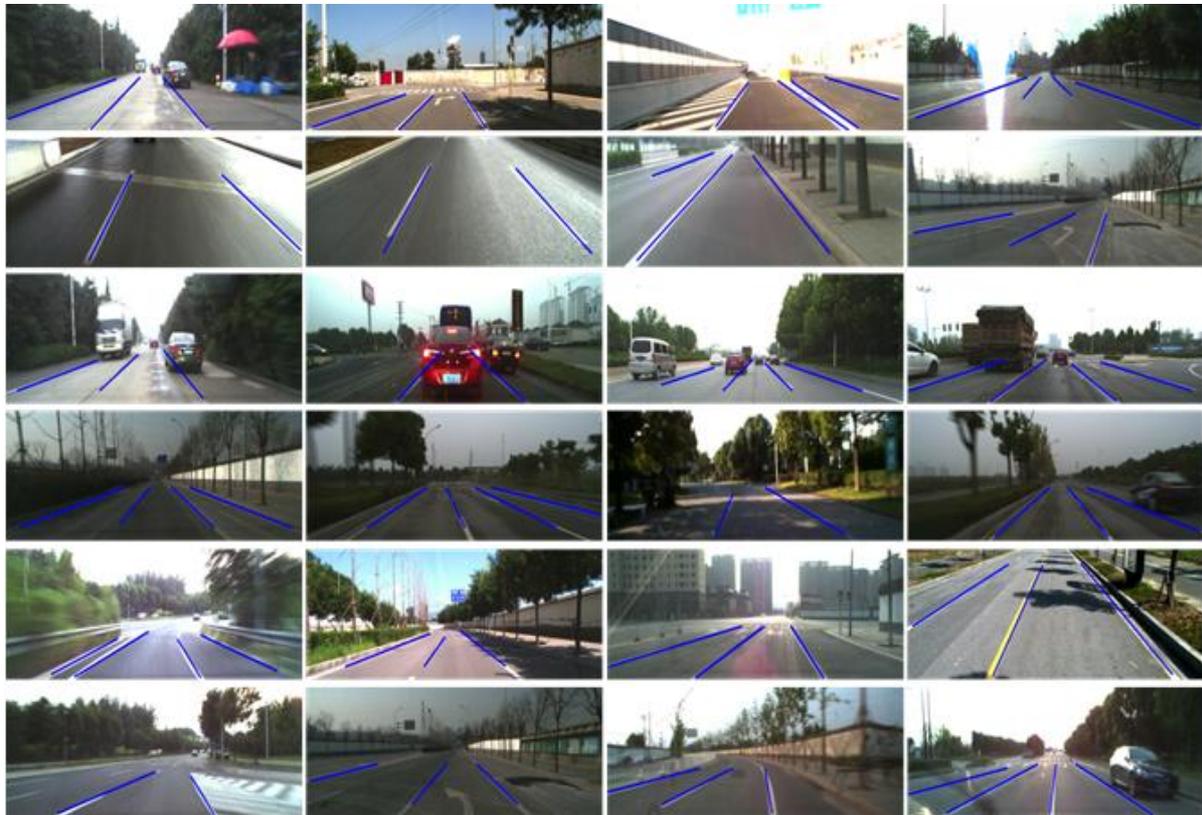


Fig.16 Results for different road conditions

CONCLUSION

In this paper, a multi-lane detection system based on RMFP for autonomous vehicle navigation in urban environment is proposed. The system works based on monocular vision and the kalman filter. Through extensive on-field tests, it has been proven that the proposed system is able to estimate the multi-lane information accurately and robustly. More importantly, the system works in real time and has the fewest limitations in practice, even works in dark nights. Its high level of accuracy and consistent good performances under difference conditions enable its implementation on the multi-lane detection for structured road.

Acknowledgement

This research was financially supported by the National Science Foundation.

The National Natural Science Foundation of China (No. 61271369, No. 61372148, No.61571045);

Beijing Natural Science Foundation (No.4152016, No.4152018);

Beijing Advanced Innovation Center for Imaging Technology. (BAICIT-2016002);

The National Key Technology R&D Program (2014BAK08B02, 2015BAH55F03)

References

- [1] Wang G, Zhou Y, Xu G, et al. A novel lane changing algorithm with efficient method of lane detection[C]// IEEE International Conference on Robotics and Biomimetics. IEEE, 2013:2458-2463.
- [2] Guo J, Wei Z, Miao D. Lane Detection Method Based on Improved RANSAC Algorithm[C]// IEEE Twelfth International Symposium on Autonomous Decentralized Systems. IEEE Computer Society, 2015:285-288.
- [3] Dahmani H, Chadli M, Rabhi A, et al. Vehicle dynamics and road curvature estimation for lane departure warning system using robust fuzzy observer [J]. *Vehicle System Dynamics*, 2015, 53(8):1-15.
- [4] Lee S, Kim S W, Seo S W. Accurate ego-lane recognition utilizing multiple road characteristics in a Bayesian network framework[J]. 2015:543-548.
- [5] J. Wang, T. Mei, B. Kong and H. Wei, "An approach of lane detection based on Inverse Perspective Mapping," in Proc. IEEE Int. Conf. ITSC, Qingdao, China, Oct. 2014, pp. 35-38.
- [6] T. Han, Y. Kim, K. Kim, "Lane detection & localization for UGV in urban environment," in Proc. IEEE Int. Conf. ITSC, Qingdao, China, Oct. 2014, pp. 590-596.
- [7] S. Jung, J. Youn, and S. Sull, "Efficient Lane Detection Based on Spatiotemporal Images," *IEEE Trans. Intell. Transp. Syst.*, vol. PP, no. 99, pp. 1-7, Aug. 2015.
- [8] J. Huang, H. Liang, Z. Wang, k Y. Song and Y. Deng, "Lane marking detection based on adaptive threshold segmentation and road classification," in Proc. IEEE Int. Conf. ROBIO, Bali, Indonesia, Dec. 2014, pp. 291-296.
- [9] R. K. Satzoda and M. M. Trivedi, "On Enhancing Lane Estimation Using Contextual Cues," *IEEE Trans. Circ. and Syst. for Video Tech.*, vol. 25, no. 11, pp. 1870-1881, Nov. 2015.
- [10] R. K. Satzoda and M. M. Trivedi, "Drive Analysis Using Vehicle Dynamics and Vision-Based Lane Semantics," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 9-18, Feb. 2015.
- [11] S.-N. Kang, S. Lee, J. Hur and S.-W. Seo, "Multi-lane detection based on accurate geometric lane estimation in highway scenarios," in Proc. IEEE IV Symp., Dearborn, Michigan, USA, June. 2014, pp. 221-226.
- [12] C. Li, B. Dai, T. Wu, Y. Nie, "Multi-lane detection in urban driving environments employing omni-directional camera," in Proc. IEEE Int. Conf. ICIA, Hailar, China, July 2014, pp. 284-289.
- [13] M. Fu, X. Wang. H. Ma, Y. Yang and M. Wang, "Multi-lanes detection based on panoramic camera," in Proc. IEEE Int. Conf. ICCA, Taichung, Taiwan, June 2014, pp. 655-660.
- [14] C. Kum, D. Cho, M. Ra and W. Kim, "Lane detection system with around view monitoring for intelligent vehicle," in Proc. IEEE Int. Conf. ISOCC, Seoul, Korea, Nov. 2013, pp. 215-218.
- [15] C. Rose, J. Britt, J. Allen and D. Bevely, "An Integrated Vehicle Navigation System Utilizing Lane-Detection and Lateral Position Estimation Systems in Difficult Environments for GPS," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2615-2629, Dec. 2014.
- [16] J. Yuan, S. Tang, X. Pan and H. Zhang, "A robust vanishing point estimation method for lane detection," in Proc. IEEE Conf. CCC, Nanjing, China, July 2014, pp.4887-4892.
- [17] V. S. Bottazzi, P. V. K. Borges and J. Jun, "A vision-based lane detection system combining appearance segmentation and tracking of salient points," in Proc. IEEE IV Symp., Gold Coast, Australia, June 2013, pp. 443-448.

- [18]V. Gaikwad and S. Lokhande, "Lane Departure Identification for Advanced Driver Assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 910-918, April 2015.
- [19]G. Liu, S. Li and W. Liu, "Lane detection algorithm based on local feature extraction," in *Proc. IEEE CAC*, Changsha, China, Nov. 2013, pp. 59-64.
- [20]M. P. Batista, P. Y. Shinzato, D. F. Wolf and D. Gomes, "Lane Detection and Estimation using Perspective Image," in *Proc. IEEE Joint. Conf. SBR LARS Robocontrol*, Sao Carlos, Brazil, Oct. 2014, pp. 25-30.
- [21]J. Wang and J. Duan, "Lane detection algorithm using vanishing point," in *Proc. IEEE Int. Conf. ICMLC*, Tianjin, China, July 2013, pp. 735-740.
- [22]U. Ozgunalp and N. Dahnoun, "Robust lane detection & tracking based on novel feature extraction and lane categorization," in *Proc. IEEE Int. Conf. ICASSP*, Florence, Italy, May 2014, pp. 8129-8133.
- [23]H. Yoo, U. Yang and K. Sohn, "Gradient-Enhancing Conversion for Illumination Robust Lane Detection," *IEEE Trans. Intell. Transp. Syst.*, vol.14, no.3, pp. 1083-1094, Sept. 2013.
- [24]Peng Pai, Wang Yaoman, Ling Zhigang, et al. Robust method for urban road detection and departure warning[J]. *Journal of electronic measurement and instrumentation*, 2015, 29(5):685-691.
- [25]Yi ShiChun, Li Keqiang, Li FangZhen, et al. A Lane Departure Warning System with Line_Based Image Procession[J]. *Automotive Engineering*, 2014, 36(9):773-778.
- [26]Wang K N, CHU X M, et al. Curved lan detection algorithm based on piecewise linear model and heuristic search[J]. *Journal of electronic measurement and instrumentation*, 2013, 27(8):689-695.
- [27]Wang K, HUANG ZH, ZH H. Algorithm for urban road detection based on uncertain Bezier deformable template[J]. *Journal of Mechanical Engineering*, 2013, 49(8):143-150.
- [28]Du X, Tan K K. Vision-based approach towards lane line detection and vehicle localization[J]. *Machine Vision & Applications*, 2015:1-17.
- [29]Hillel A B, Lerner R, Dan L, et al. Recent progress in road and lane detection: a survey[J]. *Machine Vision & Applications*, 2014, 25(3):727-745.
- [30]Assidiq A A M, Khalifa O O, Islam R, et al. Real time lane detection for autonomous vehicles[C]// *International Conference on Computer and Communication Engineering*. 2008:82-88.
- [31]Tian M, Liu F, Zhu W, et al. Vision Based Lane detection for Active Security in Intelligent Vehicle[C]// *IEEE International Conference on Vehicular Electronics and Safety*. 2007:507-511.
- [32]Truong Q B, Lee B R, Heo N G, et al. Lane boundaries detection algorithm using vector lane concept[C]// *International Conference on Control, Automation, Robotics and Vision*. IEEE, 2008:2319-2325.
- [33]Lai A H S, Yung N H C. Lane detection by orientation and length discrimination.[J]. *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*, 2000, 30(4):539-548.
- [34]Zheng B, Tian B, Duan J, et al. Automatic detection technique of preceding lane and vehicle[C]// *IEEE International Conference on Automation and Logistics*. IEEE, 2008:1370-1375.
- [35]Makiyama A. Curving tendency detection device in vehicle, and vehicle response control apparatus using same: US, US7957877[P]. 2011.
- [36]Maini R, Sohal J S. Performance Evaluation of Prewitt Edge Detector for Noisy Images[J]. 2006.
- [37]Cherri A K, Karim M A. Optical symbolic substitution: edge detection using Prewitt, Sobel, and Roberts operators[J]. *Applied Optics*, 1989, 28(21):4644-8.
- [38]Wang Y, Teoh E K, Shen D. Lane detection and tracking using B-Snake[J]. *Image & Vision Computing*, 2004, 22(4):269-280.