Lane-line Detection based on Inverse Perspective Mapping and Kalman Filter

Lijuan Zeng [1, a], Bangqian Ao [2, b], Qing Li [1, c]

[1]School of Physics and Electronic Science, Zunyi Normal University, Zunyi, Guizhou, China.

[2]College of Engineering, Zunyi Normal University, Zunyi, Guizhou, China;

[a] 1140497672@qq.com, [b] 275417559@qq.com, [c] 960755893@qq.com

Abstract. In real-time vision-based autonomous driving, lane-lines are difficult to identify due to shadows, road disrepair, reflection or other disturbances, as well as obstacles dynamic occlusion. In this paper, region of interest (ROI) is selected first. An algorithm based on inverse perspective mapping (IPM) is proposed to transform graphic images into flat images, after the operation of gray processing and image enhancement, binarization processing and secondary image enhancement are carried out for these images. Through the pixel statistics of image columns, The characteristic curves are extracted and lane-lines are fitted. At the same time, Kalman filter algorithm is proposed to predict the lane-line which are discontinuity, this improves the robustness and stability of the lane-line detection effectively. Finally, Experiments have been conducted on different scenes, the results show that the proposed algorithm own good detection efficiency.

Keywords: Lane-line detection; enhancement; kalman filter; detection efficiency.

1. Introduction

Automatic Lane-line Keeping System (ALKS) [1] becomes popular in vehicles because it could alleviate the fatigue driving of drivers to a certain extent and protect the safety of passengers. The stability of ALKS is related to the detection of lane-line closely, and it has a direct impact on the robustness of the entire control system. At the same time, it also plays an important role in the design of subsequent control systems such as Adaptive Cruise Control System (ACCS) [2]. Literature [3] designed a fast lane-line detection method based on multi-frame superposition and window search, then fitted the lane-line by selecting different lane-line models through applying the least square method. Cordts M [4] utilized Canny operator and Hough line detection to obtain various lane-lines, and designed urban car intersection background extraction and lane calibration algorithm to achieve lane-line calibration. These algorithms are to obtain the road lane-line information in real-time through monocular vision. However, due to the complexity of the environment, uneven illumination or the edge fracture and brightness discontinuity caused by obstacles, it is difficult to obtain continuous edge characteristics from the image. The real-time performance is less than optimal and the accuracy is low in scenes such as light change, road pond or shadow interference. Literature [5] [6] [7] have designed neural networks to detect the lane-lines, they could detect the lane-line accurately after training for the designed models. The main barrier hindering their universality are that the trained neural networks have high hardware requirements. Therefore, this paper designs a lane-line detection system based on inverse perspective transformation and Kalman filter algorithm. Experiments conducted in many scenes show that it has the ability of detecting lane-lines precisely and it owns the advantage of detecting various lane-lines under different environments.

2. Inverse perspective transformation

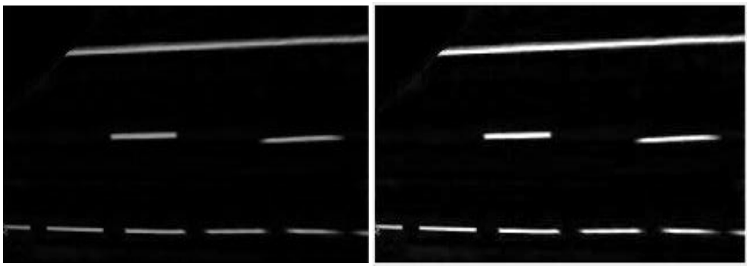
2.1 Image pre-processing

In lane-line detection, the handle for non-lane-line feature information such as sky, road signs or continuous trees beside the road would increase the complexity of calculation greatly and affect the detection accuracy. However, as a matter of fact, there is only a small part of the feature information needed where the lane-line is located. Therefore, region of interest (ROI) is utilized first. The selection of ROI could only be carried out in the fixed area for object recognition and processing. According to the position of the on-board camera, this paper selects the lower 1/2 area

image as ROI, which is shown in Figure 1.

Figure 1. Aerial view of on-board camera and ROI

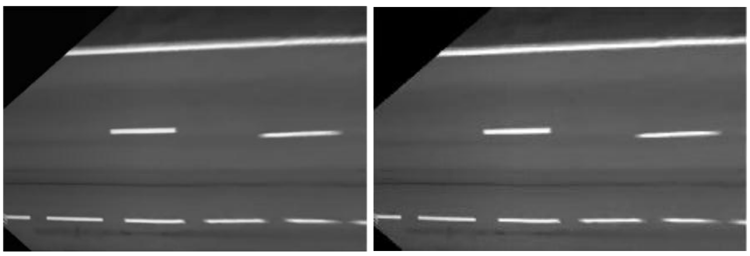
(a) (b)



(c) (d)

Figure 2. Real-time image pre-processing

The selection region of ROI reduces the unserviceable information greatly and it could improve



the real-time performance of lane-line detection effectively. The selected ROI in the image is an aerial view image, which is shown in Figure 1. Then it is converted into a 2D plane region image by inverse perspective transformation. Later, the technology of graying processing is utilized for the 2D plane region image from real-time camera which own the lane-lines, the processed image is shown in Figure 2(a).The two-dimensional Gaussian kernel function algorithm is applied to convolution the image for the purpose of strengthening the image characteristic information to achieve the goal of reducing the interference of irrelevant information. The effect of this operation is that the lane-line information has been enhanced. From Figure 2(b) we could see that the detection effect is better than Figure 2(a). In order to extract the lane line from the background image effectively, the enhanced image is segmented by binary segmentation. The result is shown in Figure 2(c), then the image is further enhanced again and the final effect is shown in Figure 2(d), from which we could see that it is better than Figure 2(b) obviously. The lane-line feature information is more distinct after the above processing, later the corresponding feature extraction operation and the fitting of the lane-line are carried out.

2.2 Lane line feature extraction and fitting

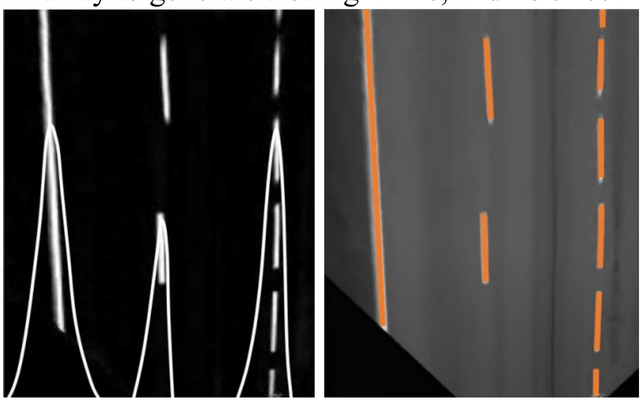
After the operation of image binarization and enhancement, the lane-line graphic image consists of the white pixel of lane-line and the black pixel of the background of non-lane line. Then operation of counting the lane points from the image center to the left and right sides is utilized, this could calculate the number of points in each column whose gray value is not zero starting from the image center to the left and right sides to count the lane points and represent it as the function $f(x)$ of abscissa x. Therefore, $f(x)$ is the extreme value of the vertical line. At the same time, as the reason of that the lane-line has a certain width and it is not vertical completely (as is shown in Figure 2 (d)), the function $f(x)$ is a fluctuating curve. Later Gaussian filter is used to combine the extreme points in order to make the sum of their extreme points as a smooth curve relatively. For those extreme points that could not be merged by Gaussian filter, α is set as the threshold. Assuming two extreme points x_1, x_2, we have:

$$x = \frac{x_1 + x_2}{2}, f(x) = \frac{x_1 f(x_1) + x_2 f(x_2)}{2} \tag{1}$$

The abscissa of the new extreme point is the midpoint of x_1, x_2, and the extreme value is the weighted average of the original two extreme points. The lane pixels of each column would be determined as a single extreme after such processing. These extreme values would be stored in the H array array in descending order for the subsequent fitting of lane-lines. For each pixel element o in H array, two points $M_1(\alpha, 0), M_1(\beta, h)$ are selected for curve fitting, and these two points meet the following constraints:

$$\begin{cases} x - b < \alpha < x + b \\ x - b < \beta < x + b \end{cases} \tag{2}$$

Where, b is the number of pixels of lane-lines width, and h representing the height of the image. If two points $M_1(\alpha, 0), M_1(\beta, h)$ are connected to generate a line, then for each pixel element o in the H array, $(2b-1)^2$ lines could be generated. If the number of pixels in a row with non-zero gray scale is taken as the statistical score, then the line with the highest statistical score is identified as the lane-line. Repeat the above process to obtain all lane-lines in the image, this operation would make each element x in the H array to generate a straight line, and its effect is shown in Figure 3.



(a) (b)

Figure 3. Pixel Calculation and Lane Extraction/ Fitting

3. Kalman lane line prediction

Kalman filter [8] is applied to track lane lines. It adopts the state space model structure of signal and noise, which updates the state vector of the system in real time through observation value. The filter estimates the random signal through the input observation and updates the current value. The updated data is used as the output signal of the filter. At the same time, the covariance is continuously recursive and the optimal value of the system is estimated. The Kalman filter prediction algorithm utilizes the real-time tracking of the object through the continuous update of the moving object in the state and time. It is a real-time recursive algorithm for the optimal estimation of the random signal. It is consists of two equations:

System state equation：

$$X(k) = AX(k-1) + BU(k) + W(k) \tag{3}$$

System measured value equation：

$$Z(k) = HX(k) + V(k) \tag{4}$$

Where $X(k)$ 、 $X(k-1)$ represent the motion state vector of the object at time k and $k-1$ respectively; $Z(k)$ indicates the state measurement vector of the system at time k ; A,B,H are three system matrices: state transition matrix, control matrix and measurement matrix; $U(k)$ represents the external input control quantity, $W(k)$ 、 $V(k)$ represent the state process noise and measurement noise of the system. In real-time application, it is usually assumed that the Gaussian white noise with zero mean value obeys the the normal distribution, namely $Noise \sim Guasian(0,\sigma)$, R 、 Q is the covariance matrix. The Kalman filter prediction algorithm mainly includes two stages of prediction and update. The formula for the system prediction stage is as follows：

State prediction formula：

$$X(k|k-1) = AX(k-1|k-1) + BU(k) \tag{5}$$

Corresponding error covariance prediction formula：

$$P(k|k-1) = AP(k-1|k-1)A^T + Q \tag{6}$$

The formula for updating the prediction results of the current state is：

State correction formula:

$$X(k|k) = X(k|k-1) + Kg(k)[Z(k) - HX(k|k-1)] \tag{7}$$

Error covariance correction formula：

$$P(k|k) = [I - Kg(k)H]P(k|k-1) \tag{8}$$

Kalman gain coefficient formula：

$$Kg(k) = P(k|k-1)H^T[HP(k|k-1)H^T + R]^{-1} \tag{9}$$

Assume the state vector is $\overline{X}_k = [x_k, y_k, v_{xk}, v_{yk},]^T$, the measurement vector $\overline{Z}_k = [x_k, y_k]^T$ represents coordinates of the centroid of the front detection object on axis x and axis y , $[v_{xk}, v_{yk}]^T$ represents the motion speed of the object on axis x and axis y . Since the interval Δt between two adjacent frames is relatively short, it is assumed that the object motion between two adjacent frames is uniform, the state transition matrix and observation matrix of the system are:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{10}$$

The covariance matrix Q and R are：

$$Q=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},\quad R=\begin{bmatrix}1&0\\0&1\end{bmatrix} \tag{11}$$

4. Determination of current lane line

After the image inverse perspective transformation and enhancement operation, the useless of Kalman filtering algorithm to predict the lane line, the integrated lane-line are detected, the effect is shown in Figure 4. However, in the non-driving lane, the lane-line detected for the subsequent control design are meaningless, such as ACCS and ALKS. According to literature [9], assuming that the detected road curvature is ρ, the lane-line heading angle is φ, the distance between the vehicle camera to the left and right lane-lines are D_{left}, D_{right} respectively, the fitting equation of the lane line can be obtained from the abscissa x of the lane line:

$$\begin{cases} y_{left} = \rho x^2 + \varphi x + D_{left} \\ y_{right} = \rho x^2 + \varphi x + D_{right} \end{cases} \tag{12}$$

Where y_{letf}, y_{right} represent the longitudinal coordinates of the left and right lane lines respectively. The lane width of the highway standards are generally 2.3~3.75 meters. In ALKS, according to literature [10], the lateral position error y_L between the vehicle and the road center line of the preview point is:

$$y_L = \left| y_{left} - y_{right} \right| \tag{13}$$

Where $y_L \in (-2m, 2\mathrm{m})$ and $\rho \in (-0.12m^{-1}, 0.12m^{-1})$, The values of y_L, ρ beyond this range are determined as non-driving lane lines, and the whole flow chart is shown in Figure 5.

Figure 4. First detection results of lane

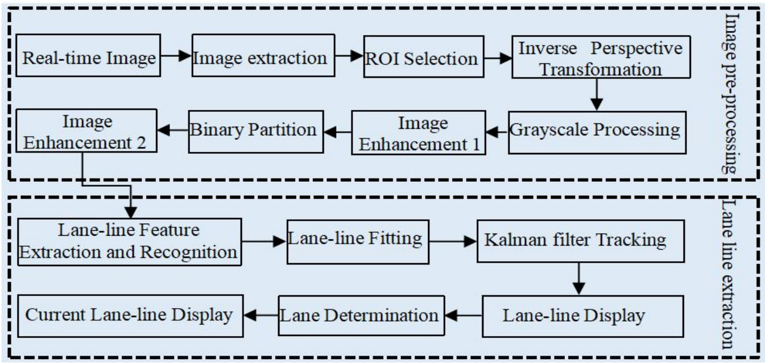


Figure 5. The flow chart of lane detection

5. Multi-area experimental test

Experiments have been conducted on three scenes, highway roads, tunnel roads and rural roads.

For highway and tunnel scenarios, the proposed algorithm works well as it is under good lighting condition. The detection accuracy of the current lane-line is high because it could obtain enough characteristic information, and the detection results are shown in Fig.6 and Fig.7. In scenario under poor lighting condition, experiments are also conducted.

In summary, the proposed method is immune to various interference and works well in various road conditions. The method has no problems like wide lane lines being identified as plurality of straight lines or dashed lanes being identified as discontinuous lines.

Figure 6. Lane detection on highway

Figure 7. Lane detection in tunnel

6. Conclusion

The accuracy of lane line detection has a very important impact on the subsequent control design. In this paper, the ROI area is selected from the graphic images obtained by the vehicle camera, and then the image information of the lane line area is obtained by using the inverse perspective transformation. After the acquired image is strengthened, the binary image is obtained by using the secondary segmentation method, and then it is strengthened once. Finally, the feature extraction and lane line fitting are performed. Kalman filter prediction algorithm is applied, This could predict their possible position in the next video image accurately.

References

[1] Xiao L Y, Gao F. A comprehensive review of the development of adaptive cruise control systems[J]. Vehicle System Dynamics. 2010, 48(10): 1167-1192.

[2] Rajamani R. Vehicle dynamic and control[M]. USA: Springer, 2006.

[3] A. Petrovai, R. Danescu and S. Nedevshi, “A stereovision based approach for detecting and tracking lane and forward obstacles on mobile devices”, in Proc. of IEEE Conference on Intelligent Vehicles Symposium (IV), pp. 634-641, June 28-July 1, 2015.

[4] Cordts M，Omran M，Ｒamos S，Ｒehfeld T，Enzweiler M，Benenson Ｒ，Franke U，Ｒoth S and Schiele B． 2016． The cityscapes dataset for semantic urban scene understanding / /Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Ｒecognition． Las Vegas: IEEE: 3213-3223.

[5] J. Kim, J. Kim, J. G. Jang and M. Lee, "Fast learning method for convolutional neural networks using extreme learning machine and its application to lane detection," Neural Networks, vol. 87, pp. 109-121, March, 2017.

[6] J. Li, X. Mei, D. V. Prokhorov, D. Tao, Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene. IEEE Trans. Neural Netw. Learning Syst., vol. 28, no.3, pp. 690-703, 2017.

[7] J. Kim, C. Park, End-To-End Ego Lane Estimation Based on Sequential Transfer Learning for Self-Driving Cars. CVPR Workshops, pp. 1194-1202, 2017.

[8] L. Chanho, M. Jihyun, Robust Lane Detection and Tracking for Real-time Application[J]. IEEE Transaction on Intelligent Transportation Systems, 19(12): 4043-4048, 2018.

[9] X. Du, K. Tan, Vision-based Approach Towards Lane Line Detection and Vehicle Localization[J]. Machine Vision and Application, 27:175-191, 2016.

[10] R. Fan, N. Dahnoun, Real-time Stereo Vision-based Lane Detection System[J]. Measurement Science and Technology, 29(7): 1-24, 2018.