Research on Detection Algorithm of Roadway Line on Structured Road Based on Vision

Suke Qian, Jie Yang*, Chao He and Zonghao Li

School of Machinery and Transportation, Southwest Forestry University, Kunming, Yunnan 650224

*Correspondence author: jieyang2020@swfu.edu.cn

Abstract. To solve the problem of low efficiency of lane recognition in different structured road environments, an improved Hough transform algorithm is proposed. Firstly, every frame of image in the video stream is processed by gray scale and inverse perspective transformation, so as to reduce the amount of data and improve the accuracy of lane recognition. Then, median filtering and wavelet hard threshold denoising are combined to remove noise and improve image quality. Secondly, the image is binarized and edge detected by OTSU algorithm and Sobel operator, and the lane feature information is extracted. Finally, the traditional Hough transform algorithm is improved, and the slopes and points of the selected straight lines in the feature area are averaged, and then a new straight line is obtained and displayed in the image. Simulation experiments show that the algorithm has good applicability and reliability in different structured road environments, and achieves the expected experimental results.

Keywords: Machine vision, Image filtering, Lane line recognition, Hough transform.

1. Introduction

The development of science and technology, intelligent assisted driving technology has developed rapidly. Because of the advantages of machine vision system, such as low cost, multi-purpose, wide information coverage, simple structure, convenient operation, etc [1]. Unmanned intelligent assisted driving based on vision has become a hot topic in this field.

At present, the application environment of vision-based unmanned intelligent assisted driving system is mostly structured road, which is safe to carry out the important premise of driving is how to identify the car correctly. Road lines and provide navigation signals for vehicles; Qian Yi [2] finds the feature area by preprocessing the image, and then detects the lane line by Hough transform. The algorithm has high performance in highway environment, but it is not suitable for complex road environment; Cheng et al. [3] separated the lane line from the background by linear analysis of the background environment and RGB channel of the lane line in the test environment of the expressway, thus extracting the lane line. However, this method has a great error in the detection effect of lane lines under the influence of external factors; Bi Yanbing et al. [4] passed the LMedSquare curve the fitting technology is used to detect the lane line of urban structured roads. The algorithm has good robustness, but the calculation is large and the real-time performance can not meet the requirements.
To sum up, the vision-based unmanned intelligent assistant driving system needs not only real-time and reliability, but also the performance requirements of accurate recognition in complex environment when detecting lane lines on structured roads. Therefore, this paper proposes an improved and effective lane detection algorithm to improve this situation. Its work mainly includes the following steps: Firstly, preprocessing measures are taken for images collected by vehicle-mounted cameras, mainly to reduce the amount of image data, enhance image information, filter out image noise and improve image processing speed. Secondly, the edge features of lane lines are extracted to prepare for lane line fitting; Thirdly, the lane line fitting is carried out to provide navigation signals for the subsequent driving of the vehicle.

2. Image preprocessing
The initial image collected by the vehicle-mounted camera is not suitable for direct detection because it contains noise or the amount of data is too large. Otherwise, it will affect the real-time performance of data processing, resulting in a large error and a decline in the accuracy of lane line recognition.

In fact, the video stream collected by the car camera is composed of frames of images, each of which is a three-dimensional image as shown in Figure 1, and the amount of data is very large. The original three-dimensional image is changed into a one-dimensional image by graying each frame of image, and the data amount is reduced to the original, which greatly improves the image processing speed; Then, in order to improve the accuracy of trace recognition, the inverse perspective transformation operation is carried out Finally, in order to reduce the influence of noise and reduce errors, the image is filtered and denoised. The specific flow of image preprocessing is shown in Figure 2.

2.1. Selection of grayscale algorithm
Nowadays, the main methods of image grayscale are component method, maximum method, average method and weighted average method [5]. After analysis, the author uses weighted average method to gray the image, and the gray image is shown in Figure 3. The weighted average method is adopted because the four methods introduced treat the three components equally, but the first three methods do not consider the importance of the proportion of different components, while the weighted average graying gives weight to the three components, and then takes the weighted average value as the gray value, and its graying formula is as follows:

\[ f(i,j) = 0.299R(i,j) + 0.587G(i,j) + 0.114B(i,j) \]

The former numbers are the weights assigned to different components.

Figure 1. Color (RGB) images.
2.2. Inverse perspective transformation
In the collected image, the shape of the road route formed from vanishing point to close range can be approximately regarded as trapezoid, which will cause a large error in the detection result of the distant lane line. By using the inverse perspective transformation \[6, 8\], the vehicle coordinates are transformed into world coordinates, and the image is transformed into a bird's eye view. The image after inverse perspective transformation is shown in Figure 4. In the original image, the lane line in the distant view is clearly displayed without distortion or missing, which is basically consistent with the real lane line, making it easier to screen the slope of the subsequent straight line and filtering out most irrelevant information (such as surrounding trees, route signs, etc.).

2.3. Filter smoothing
In fact, due to the intensity of light, vehicle turbulence, camera imaging channel instability and other factors will lead to the acquisition of images with noise, in order to reduce the noise in the image and enhance the image quality, the image must be de-noise smoothing processing. In addition, the noise carried by images is often mixed with various kinds. In this case, if a filtering algorithm is simply adopted for noise reduction, the noise is only partially suppressed and the effect of noise removal is not good. Therefore, median filtering and wavelet hard threshold denoising are combined in this paper. Fig. 5 (a) is the curve of a line extracted from the image containing the grayscale value of the lane line pixels, and Fig. 5 (b) is the grayscale curve after noise reduction. After comparison, it is proved that the method adopted in this paper meets the requirements of noise reduction.
3. Edge feature extraction

This module can be divided into two parts: binary image processing and binary image edge detection. The binarization is to separate the lane line from the background. Edge detection is to highlight the feature information of the edge.

3.1. Image binarization

Image binarization is to find a threshold, which can separate the object from the background. In this paper, the gray histogram is compared with the threshold table obtained by different binarization methods, and the most suitable method is selected. Table 1 shows the thresholds obtained by different binarization methods, in which the thresholds obtained by the minimum error method and the global threshold method are located at points A and B in the gray histogram of Figure 6, and the thresholds are too small to separate the targets and produce new noises. The threshold values obtained by iterative method and Otsu method are located at C and D points in gray histogram, and the threshold values are reasonable, but the iterative method has too much computation. Based on the research background of this paper, OTSU image binarization method is adopted, and the binarized image is shown in Figure 7.

![Gray level curve of the original image](image1)

![Gray curve after noise reduction](image2)

Figure 5. The gray curve of the image.
3.2. Edge detection
Image edge detection algorithms commonly used include: Canny operator, Prewitt operator, Roberts operator and Sobel operator [9]. However, when the Canny operator deals with the road with tree shadow, it will also extract the contour of the tree shadow, which will lead to the increase of error [10]. Prewitt operator is easy to misjudge edge points [11]. Roberts cannot suppress the shadow of noise [12]; Sobel operator can suppress noise, and it can also select horizontal or vertical edges for extraction [13]. In order to accurately propose the edge information of the lane line and reduce the error, the Sobel operator is selected in this paper to detect the edge feature information, and the effect is shown in Figure 8.

| Threshold value | Iterative method | OTSU | The minimum error difference method | Global threshold Value method |
|-----------------|------------------|------|------------------------------------|------------------------------|
|                 | 130              | 124  | 78.999                             | 79.924                       |

**Figure 6.** Gray histogram.

**Figure 7.** Binarization image.
6

Figure 8. Sobel edge detection.

Figure 9. Hough straight line detection.

4. Fitting of lane lines

After image graying, inverse perspective transformation, image noise reduction, binarization and edge detection, lane lines are successfully extracted. What needs to be done next is to identify and represent the extracted lane lines in the original image.

Generally, the industry standard for the design and construction of structured roads is that the limit turning radius is 650m when the speed is 120km/h, and the general minimum radius is 1000m [13]. When driving, the nearby lane line can be approximated as a straight line. The traditional method is to use the straight line model as the lane line model and identify the lane line by Hough transform algorithm.

Although Hough transform has high efficiency in line detection, its application still has some limitations. As shown in Figure 9, if the route we want is from point A to point B, it is easy to detect the line by Hough transform in the case of (a), but it will detect and three routes in the case of (b), with a large error. In view of this situation, this paper improves the traditional algorithm. In this algorithm, the lane line area after edge detection is set as the feature area by programming in advance, and then
the feature points in the whole area are searched for extremum, and the straight lines with large slope difference are eliminated to improve the recognition accuracy. Next, the slope of the straight line detected in the feature area and all the points \((x, y)\) in the straight line equation are averaged; According to the formula, a new intercept is obtained by using the average slope and average point obtained in the previous step; Finally, a new straight line is determined by slope and intercept, which is highlighted on the bird's eye view, and then inverted to the original image by using the inverse perspective transformation matrix, so that the recognition effect of lane lines in the original image can be realized, as shown in Figure 10(a) and (b).

4.1. Algorithm flow

The algorithm in this paper has high accuracy. Because the data volume is reduced and the image information is enhanced by preprocessing and feature extraction in the early stage, the algorithm also has high real-time performance and robustness. The algorithm flow is shown in Figure 11.

4.2. Experiment

Structured road environment can be divided into two types: structured roads in expressway environment and structured roads in urban road environment. The former has clear lane lines and no serious interference factors, while the latter has complex road environment and many interference factors. In this paper, recognition experiments are carried out on these two environments to verify the performance of the algorithm. From Figure 11, we can see that the lane line recognition effect is very good in the expressway environment, and the next step is the test in the urban road environment.

(a) The effect of traditional algorithm

(b) The algorithm effect of this paper

Figure 12. Comparison of algorithm recognition effects.
The pictures in Figure 12 are all taken by the author: (a) The lane line is lightly worn and the interference factors of the road environment are single; (b) It is a road environment with serious wear and tear on the lane line, cracks and pits on the road surface and sundries on the lane line; (c) It is a road environment with severely damaged lane lines and roads. It can be seen from the experimental renderings that the improved algorithm has higher recognition accuracy and stronger applicability than the traditional algorithm in the complex road environment.

4.3. Analyses of results
Through the previous simulation experiment, the feasibility and accuracy of the improved algorithm have been verified. However, compared with the traditional algorithm, whether its comprehensive performance is improved needs to be further analyzed. The following two charts compare the results respectively in the freeway environment and in the urban road environment. The hardware environment of the algorithm is Core i5-7200U, 3.1Hz, Win10 system, 4G memory; The software environment is MATLAB 2020A.

The total number of test patterns is 130. Table 2 shows the comparison of lane line identification data in expressway environment. Table 3 shows the comparison of lane line identification data in urban road environment, in which, as shown in Fig. 12, there are 40 road condition images (1), 50 road condition images (2) and 40 road condition images (3).

| Table 2. Comparison of highway environmental results. |
|------------------------------------------------------|
| The average time (t/s) | Total time (t/s) | Accuracy (%) |
|------------------------|------------------|--------------|
| Traditional algorithm  | 0.041s           | 5.31s        | 98.4%        |
| This text algorithm    | 0.040s           | 5.29s        | 99.2%        |

| Table 3. Comparison of urban road environmental results. |
|--------------------------------------------------------|
| The average time (t/s) | Total time (t/s) | Accuracy (%) |
|------------------------|------------------|--------------|
| Traditional algorithm  | 0.043s           | 5.59s        | 93.8%        |
| This text algorithm    | 0.041s           | 5.32s        | 97.7%        |

It can be seen from the above table that the accuracy of lane line recognition in expressway environment is slightly higher and the processing time is basically the same; When the scene is transformed into urban road, the algorithm in this paper has higher accuracy and shorter processing time than the traditional algorithm. Through the knot after analysis, the comprehensive performance of the improved algorithm for lane detection in structured road environment meets the expected experimental results.

5. Conclusions
The author improves the image preprocessing stage to reduce the amount of image data, so as to meet the real-time requirements of vehicles; In the part of lane line fitting, the traditional algorithm is improved and verified by experiments. Experiments show that the proposed algorithm is more suitable for recognition in some complex environments, and improves the accuracy and robustness; The experimental results of the algorithm in this paper are satisfactory on the whole, but the effect of lane line detection at road bends is not perfect, which is caused by taking the whole lane line as a straight line model and neglecting the curvature to a certain extent, which is also the place that needs to be improved next.

References
[1] JIN H, WU L L, CHEN H Y, GONG J W. An Improved Algorithm for Lane Line Recognition of Structured Road [J]. Journal of Beijing Institute of Technology, 2007 (06): 501-505.
[2] QIAN Y. Lane line Detection based on Hough Transform [J]. Shandong Industrial Technology,
2017 (10): 272-273

[3] Cheng H Y, Jeng B S, Tseng P T, et al. Lane detection with moving vehicles in the traffic scenes [J]. IEEE Transactions on Intelligent Transportation Systems, 2006, 7 (4): 571-582.

[4] BI Y B, GUAN X, ZHAN J. A method for searching lane lines during lane identification [J]. Automotive engineering, 2006, 28 (5): 439-442.

[5] WANG Y Z, WANG X Y, WEN C L. Gradient point to Constraint detection of structured lane [J]. Chinese Journal of Image and Graphics, 2016, 17 (6): 657-663.

[6] HE H X, CHANG L, JIANG H T, ZHANG Q Y. An adaptive lane line detection method based on clustering inverse perspective Transformation [A]. China Society of Automotive Engineers. Proceedings of the annual Conference of 2020 (1) [C]. China Society of Automotive Engineers, 2020: 4.

[7] ZHONG Z B. A graying method for lane line recognition [J]. Journal of tongji university (nature science edition), 2019, 47 (S1): 178-182.

[8] GAO D Z, ZHENG B G, DUAN J M. Intelligent vehicle positioning technology based on inverse perspective transformation [J]. Computer measurement and control, 2009, 17 (09): 1810-1812.

[9] CHANG N. Review of edge detection Algorithms in Image processing [J]. China Science and Technology Information, 2011 (04): 130-131+149.

[10] LI C Y, CHEN G X, DING Y J. Improved edge detection algorithm for Canny operator [J]. Miniature microcomputer system, 2020, 41 (08): 1758-1762.

[11] S. A. Fakhri, S. A. Fakhri, M. Saadatseresht. ROAD CRACK DETECTION USING GAUSSIAN/PREWITT FILTER. 2019, XLII-4/W18:371-377.

[12] WANG F C, ZHANG M, GONG L M. Improved Roberts image edge detection algorithm [J]. Journal of detection and control, 2016, 38 (02): 88-92.

[13] ZHENG H H, BAI Y X, ZHANG Y Q. An edge detection algorithm based on Sobel operator [J]. Application of microcomputer, 2020, 36 (10): 4-6.