An Improved Hough Transform Method for Detecting Forward Vehicle and Lane in Road

Yingsheng Wang*, Yuhan Qi, Yi Man
China North Vehicle Research Institute, Beijing 100072, China

*yingshengw@163.com

Abstract. In this paper, we propose an integrated method for lane detection and vehicle detection, which tries to make a real-time analysis of vehicle video for identifying lane, detecting, and tracking forward vehicles. In lane detection, image preprocessing, line detection based on improved Hough transform, and straight-line model reconstruction are used. For vehicle detection, preprocessing, vehicle shadow merging based on the improved search algorithm, regions of interest (ROI) demarcation, lane determination, and vehicle tracking are used. The experiment results show that the time it takes to process an image is about 25ms. Additionally, the lane detection rate of vehicles driving on a structured road is approximately 98%, and the vehicle detection rate of the closest forward vehicle is approximately 81%.

Keywords: Computer vision, Driver assistance, Lane detection, Vehicle detection

1. Introduction

On the one hand, the sky-rocket increment of vehicle numbers brings some convenience to people; on the other hand, it inevitably causes traffic congestion, frequent accidents, and other problems. Over the past decade, countries worldwide regard the Advanced Driver Assistance System (ADAS) as the key to solve this problem.

In an automatic vehicle, ADAS plays a vital role. With the continuous progress of the automobile industry [1], ADAS can use various sensors to detect the surrounding environment and collect information. ADAS can use this information to assist in environmental perception and vehicle behavior analysis. For example, low-resolution lidar sensors are used for front collision avoidance [2]. The detection method based on video image processing absorbs the advantages of other detection methods. For lane detection, the performance evaluation method is used to analyze and compare two edge detection methods of Canny and Sobel in [3]. In recent years, the use of deep learning has become a hot direction in the field of lane detection. [4-6] use depth neural network to train data set and extract lane features. For vehicle detection, [7] uses a combination of two HOG vectors to extract vehicle features. Lane detection and vehicle detection have always been an active research field, but few studies explore the integration of these. The research showed that the combination of these could improve the detection rate of vehicles. In the earlier study, as in [8], a probabilistic data association filter is used to detect lane and vehicle. However, lane and vehicle detection performance is not explicitly verified. Lane tracking and vehicle tracking are combined to improve lane location performance as in [9]. Whereas, the relevant information is not used to improve vehicle detection. Although [8] and [9] realize the integration of lane and vehicle detection to some extent, they do not
show that the integration can significantly improve the whole system’s performance. In recent years, both [10] and [11] have achieved specific results in the comprehensive detection method of lane and vehicle.

In this paper, we propose a forward vehicle detection method, combines lane detection and vehicle detection, that can extract distance and speed information of the forward vehicle. The lane detection is based on improved Hough transform and vehicle detection applies an improved search algorithm. The proposed method analyses the video sequence frame by frame. Lane detection and vehicle detection are carried out sequentially. Lane detection is realized by three steps: image preprocessing, line detection based on improved Hough transform, and straight-line model reconstruction. Vehicle detection is realized by six steps: preprocessing, vehicle bottom shadow merging based on the improved search algorithm, regions of interest (ROI) demarcation, lane determination, and vehicle tracking. Meanwhile, the method makes some improvements.

2. Lane detection methodology
In this method, the video sequence is analyzed frame by frame. We divide the lane detection into three parts to introduce: image preprocessing, line detection based on improved Hough transform, and linear model reconstruction.

2.1. Image preprocessing
Image preprocessing includes image graying, smooth denoising, and edge detection. The purpose of these three methods is to reduce the useless information and the noise in the image. First, we choose to use the library functions that come with OpenCV. Next, smooth denoising is done. Finally, we select Canny edge detection with strong anti-jamming ability and suitable for complex noise images. Fig. 1 (a) is the original image. Fig. 1 (b) is the gray image. Fig. 1 (c) is the edges image.

![Fig. 1 (a) Origin image, (b) Gray image, (c) Edges image.](image)

2.2. Line detection based on improved Hough transform
Hough transform is a feature extraction technique in image processing. It uses a voting algorithm to detect the specific shape of objects. Usually, Hough transform needs to search the whole image, which requires a large amount of computation. Therefore, this paper improves the Hough transform and comes up with our own method of improved Hough transform algorithm.

Generally speaking, the left lane and the right lane in the vehicle camera should be located in the left and right half of the image respectively. The lane's position and angle will not change suddenly when the vehicle is driving in a straight line. We calculate the location of the lanes in different environments and get the range of the lanes. When using Hough transform to detect straight lines, we only need to detect in this range, and the search range can be reduced. According to statistics, the lane is generally located at -30°~150° in the image coordinate space. According to this characteristic, we can remove the edge detection results outside this range and remove many interference points that do not belong to the lane, such as buildings, trees, and other background interference points. Finally, we reduce the computational complexity of the Hough transform to the original 1/3. Fig. 2 (a) shows the effect before processing. Fig. 2 (b) is the effect after processing.

2.3. Linear model reconstruction
By performing the Hough transform within the filtered range, the result is a set of points representing each straight line, rather than the final required lane. Thus, we need to use the linear model to reconstruct the lane.

![Fig. 2 (a) Effect before processing (b) Effect after processing](image)

First of all, the slope of the straight line represented by every two points is calculated. If the slope is negative, it is classified as the left lane, and if the slope is positive, it is classified as the right lane. Then the slope of the two groups of lines is averaged. Next, we calculate the difference between each line's slope and the average slope. Suppose the difference is more significant than the threshold. In that case, it can be considered that this line is an unrelated interference line in lane detection. After this screening, the points left are regarded as points on the real lane. Finally, the left and right lanes' points are fitted by the least square method of the first-order polynomial, and the final lane to be detected is obtained. Suppose the coordinate of a point is \((x, y)_i\), we can get the fitting line according to (1),

\[
\begin{align*}
    a &= -\frac{\sum_{i=1}^{n} x_i y_i - nx y}{\sum_{i=1}^{n} x_i^2 - n x^2} \\
    b &= \frac{\sum_{i=1}^{n} x_i y_i - n x y}{\sum_{i=1}^{n} x_i^2 - n x^2}
\end{align*}
\]

where \(a\) represents the intercept of the line, and \(b\) represents the slope of the line.

3. Vehicle detection methodology
Vehicle detection is realized by six steps: preprocessing, vehicle bottom shadow merging based on the improved search algorithm, ROI demarcation, lane determination, and vehicle tracking.

3.1. Preprocessing
Using the OpenCV extension library in Python, the image can be directly transformed from RGB space to HSV space. Then the HSV color value of vehicle bottom shadow is selected through practical experiments. Three image sequences from the vehicle camera's video are selected, from which six shadow points are extracted. And the corresponding values of \(H\), \(S\), and \(V\) are calculated by Python. Table 1 shows the result.

| Points | Values of \(H, S, V\) |
|--------|----------------------|
| Point 1| 177, 97, 55          |
| Point 2| 2, 105, 46           |
| Point 3| 176, 233, 23         |
| Point 4| 4, 122, 46           |
| Point 5| 5, 124, 41           |
| Point 6| 2, 151, 32           |

From Tab. 1, the range of detected range in the HSV color space is selected as lower = [0, 90, 10], up-per= [180, 255, 60].
Next, it is necessary to modify the shaded areas to connect the damaged areas and separate the adhesive areas. In this paper, the open operation in morphological processing is used to deal with it. We select a horizontal convolution core with 11 pixels long and 1 pixel wide to ensure that isolated noise points and interference lines with horizontal length less than 11 pixels can be eliminated.

3.2. Vehicle bottom shadow merging based on the improved search algorithm

After morphological processing of the image, the shadow extracted from the vehicle's bottom usually appears as a block. Therefore, it is necessary to merge the shadow so that it will eventually become a line.

To extract the real bottom shadow, the image is retrieved in the y-direction from top to bottom. Assuming that \( y_1 \) and \( y_2 \) respectively represent the position of two shadow lines of the image, the merging principle of adjacent shadow lines is

\[
|y_1 - y_2| < TH
\]

(2)

According to the analysis of the experimental results, the distance between the upper and lower lines merged by \( TH=10 \) pixels, which can meet the requirements. In merging two adjacent shadow lines, the merging can be carried out by following the principle of finding the leftmost endpoint, the rightmost endpoint, and the maximum y value.

Three things need the Lord:

- During vehicle tracking, the vehicle tracking range is artificially selected in the first frame of the video sequence to initialize the tracker. The tracker will automatically generate each subsequent frame tracking range. The range is dynamic, so the algorithm deals with this dynamic range.
- An improved search algorithm is used to search for the selected area of vehicle tracking. The image is scanned line by line from top to bottom in the y-direction. Then an interval of 8 pixels is applied from left to right in the x-direction. Since a structural element with a length of 11 is selected for processing during the open operation, the 8-pixel interval will not miss any shadow lines.
- Suppose \( f(x, y) \) is a pixel in the image. If a point satisfies (3) for any row, that means a white pixel is found. Then we search from that point to the left and right sides pixel by pixel. If (4) is satisfied, we mark the point as the starting point \( x_{start} \). If the (5) is satisfied, we mark the point as the ending point \( x_{end} \).

\[
\begin{align*}
   f(x, y) - f(x-8, y) &= 255 \\
   f(x, y) - f(x-1, y) &= 255 \\
   f(x, y) - f(x+1, y) &= 0 \\
   f(x, y) - f(x-1, y) &= 0 \\
   f(x, y) - f(x+1, y) &= 255
\end{align*}
\]

(3) (4) (5)

3.3. ROI demarcation

After the above merging, there are only a few horizontal shadow lines left in the image. They are generally formed by the forward vehicle, and these shadow lines all represent the areas where the vehicle may exist. We need to establish the ROI areas of the vehicle to confirm further.

We can set the threshold according to the perspective projection relationship. If a detected shadow line is within the threshold range, it can be considered the forward vehicle that needs to be detected in this paper. Otherwise, it will be screened out.

The projection width of the vehicle in the image can be approximately represented by (6),

\[
w = \frac{w_{exp}}{H} y
\]

(6)

where \( w \) represents the target object's length in the image, \( w_{exp} \) represents the actual width of the forward vehicle, \( H \) represents the installation height between the vehicle camera and the road, \( y \) represents the pixel line of the target object in the image coordinate system.
The process of restricting the ROI area is as follows.
Step 1, we need to calculate the length $l$ of each shadow line according to (7).

\[ l = x_{end} - x_{start} \]  

(7)

Step 2, we filter the shadow lines according to (8).

\[ 0.48 \times w < l < 1.2 \times w \]  

(8)

Step 3, represents the height of the shadow line to the top of the image and represents the image height. According to (9), vehicles' approximate height with different distances in the image can be obtained.

\[ h = \frac{H_{pin} - l}{H_{msg}} \]  

(9)

Finally, we can delimit the ROI region.

3.4. Lane determination
The one we need needs to be identified from many ROI areas. After analysis, a criterion of the target vehicle can be obtained: The target vehicle's ROI area always intersects the detected lane. Therefore, it can be determined whether the ROI area intersects the lane. If it intersects, it is considered to be the target vehicle. If it does not intersect, it is considered to be an unrelated vehicle.

3.5. Vehicle tracking
Due to the limited computing power of on-board equipment, it is unreasonable to search for each image for vehicle detection. The frame image object has an excellent position correlation, which can be used to predict the next frame's target position. This paper chooses to call the MOSSE tracker in the OpenCV extension library to reduce the algorithm's processing range and speed up the processing speed.

4. Experiments and results analysis
To verify the method proposed in this paper, we have tested various conditions with actual vehicles on the highway. As a test-bed in this respect, we used high-resolution driving records installed above the rearview mirror of a standard passenger car to take images. In order to measure the performance of the proposed method, the following experiments are carried out. All the images used in this experiment are taken from the actual road, and the image size is 960 to 540 pixels. In order to detect lane features from actual road images, a total of 2000 images were obtained. The forward vehicle in the image is undergoing irregular acceleration and deceleration.

The hardware platform of this method is mainly composed of a vehicle camera and a visual inspection computer. The type of monocular vision sensor used in this method is AKASO Dash Cam V1. It is communicated with the visual inspection computer through a USB data cable. In this paper, the resolution is 1080p, the frame rate is 30fps, and the horizontal viewing angle is 170 °. The visual inspection computer processor uses Inter (R) Core (TM) i7-9750H CPU @ 2.6GHz processor, 16.00GB memory, Windows 10 64-bit operating system. The main content of the software in this paper is Python and the OpenCV extension library. The version of Python is 3.7.7 and the version of the OpenCV extension library is 4.2.0.34. Fig.3 shows the final effect.

![Fig. 3 Two situations of vehicle detection](image)

Fig. 3 Two situations of vehicle detection
We selected a small segment of the video sequence to illustrate the effectiveness of the method. In the case of traditional Hough transform and traditional search algorithm, the algorithm processes the whole image. The average time required to process an image is 1.68s. And due to the influence of a large number of interferences, the algorithm can't accurately detect lanes and vehicles. In the case of our method, the success rate of lane detection is more than 98%, and the success rate of vehicle detection is more than 81%. The average time required to process an image is about 25ms. After comparison, we can see that the improved algorithm greatly improves the detection speed and performance. In most cases, the failure of vehicle detection is due to the long distance.

5. Conclusion and promising research
This paper's main content is to design a forward vehicle detection method combining lane detection based on improved Hough transform and vehicle detection based on the improved search algorithm. Constructive improvements have been made in many aspects, such as Hough transform, search algorithm, and search scope. Experimental results show that our method can accurately extract the forward vehicle's position and speed information based on fast and straightforward. In the future, we will do the following research to enrich the system: first, research on unstructured roads; second, research on the error in vehicle tracking; third, research on extreme cases.

References
[1] Clara Marina M, Mira H, Fei-Yue W, Bo G and Dongpu C 2018 Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey IEEE Transactions on Intelligent Transportation Systems 19 (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp 666-676
[2] Qian L, Yurui C, Jiajia L and Abderrahim B 2019 Localization and navigation in autonomous driving: Threats and countermeasures J. IEEE Wireless Communications 26 (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp38-45
[3] Sunil Kumar V, Akash and Divakar Singh Y 2015 Analysis of lane detection techniques using openCv J.12th IEEE International Conference Electronics, Energy, Environment, Communication, Computer, Control: (E3-C3), INDICON 2015 (Piscataway: Institute of Electrical and Electronics Engineers Inc)
[4] Ravi B, Akshay Uttama N, Venkata N. P and Bhaskaran R 2020 Driving lane detection on smartphones using deep neural networks ACM Trans. Sens. Netw 16 (New York: Assoc Computing Machinery) p22
[5] Wei W, Hui L and Junshu W 2020 Cnn based lane detection with instance segmentation in edge-cloud computing J. Cloud Comput.-Adv. Syst. Appl 9 (New York: Springer) p10
[6] [Qin Z, Hanwen J Qiyu D, Yuanhao Y, Long C and Qian W 2020 Robust lane detection from continuous driving scenes using deep neural networks IEEE Trans. Veh. Technol 69 (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp41-54
[7] Gang Y, Ming Y, Yang Y and Longfei F 2016 Real-time vehicle detection using histograms of oriented gradients and adaboost classification Optik 127 (Munich: Elsevier GmbH) pp7941-7951
[8] Ssu-Ying H, Yi-Ming C, Bin-Feng L, Li-Chen F, Pei-Yung H and Shin-Shinh H 2009 Tracking and detection of lane and vehicle integrating lane and vehicle information using pdaf tracking model IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp 603-608
[9] Sayanan S and Mohan Manubhai T 2010 Improved vision-based lane tracker performance using vehicle localization IEEE Intelligent Vehicles Symposium, Proceedings (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp676-681
[10] Quang Van N, MoonYoung Y, WooSeong C, DukSun Y, HeungSeob K and KwangSuck B 2016 A study on real time integrated lane detection and vehicle tracking method with side-
mirror cameras 2016 *IEEE 14th International Workshop on Advanced Motion Control AMC 2016* (Piscataway: Institute of Electrical and Electronics Engineers Inc) pp346-352

[11] Van-Quang N, Changjun S, Heungseob K and Kwangsuck B 2018 A study on detection method of vehicle based on lane detection for a driver assistance system using a camera on highway *2017 Asian Control Conference ASCC 2017*(vol 2018-January) (New York: IEEE) pp424-429