

FAST LANE TRACKING USING HMM AND MHT FOR AUTONOMOUS URBAN DRIVING

KEMAL KAPLAN, CANER KURTUL and H.LEVENT AKIN

*Department of Computer Engineering, Boğaziçi University
Istanbul, Turkey*

Lane tracking is one of the most important processes for autonomous vehicles because the navigable region usually stands between the lanes, especially in urban environments. Hough transform is one of the most popular algorithms for lane detection. A robust lane tracking method is also required for reducing the effect of the noise and the required processing time. In this paper, we present a new lane tracking method which uses a partitioning technique for obtaining multiresolution Hough Transform (MHT) of the acquired vision data. After the detection process, for tracking the detected lanes, a Hidden Markov Model (HMM) based method is proposed. This work is a part of the ADES (Automatic Driver Evaluation System) Project in Artificial Intelligence Laboratory of Boğaziçi University.

1. Introduction

Although the vehicle manufacturers deploy more intelligence in their newest models, the current applications are usually focused on driver assistance and early warning systems. However, in the near future, intelligent vehicles will also enforce the traffic regulations. For example, speed limit and traffic light violations are going to be detected by cars. With this motivation, the aim of the ADES project is to develop a framework for evaluating the drivers against the traffic rules. The applications of the resulting system include but not limited to the following items;

- Deter drivers from violating traffic rules
- Automation of driver license examinations
- Supervising the development of autonomous urban driving

The project can be divided into two major parts. The first part is acquiring the necessary information from various sensors where the second part is processing these knowledge to evaluate the driver's actions.

In this work we present a new approach for obtaining lane boundaries from vision data. The proposed approach uses MHT for detecting lane markers followed by an HMM model for tracking them.

2. Related Work

For lane detection, Hough Transform [1] is one of the most common techniques [2, 3, 4]. However, there are many other techniques in the literature for lane detection. Pomerleau et al. [5] used neural networks in their placeALVIN system. Dynamic programming is used for eliminating outliers from detected line segments by Kang et al. [6]. In addition, Wang et al. [7] used B-splines in order to fit lane markings. Different techniques have also been proposed for tracking the detected lanes and modeling the road. Kalman filtering [8] and particle filtering [9, 10] are the two most common tracking techniques used in lane tracking methods. A more detailed survey for lane detection strategies can be found on [11].

3. Proposed Approach

The proposed approach employs MHT for lane detection, followed by two HMM models for radius and orientation of the candidate lanes.

3.1. *Multiresolution Hough Transform*

The classical Hough transformation approach processes the entire vision data in order to detect the lines. This scenario has two main drawbacks. First, the occluded lines (i.e. another car passing through the line) become noisy since the transformed relative intensity of the line decreases. Second, the relative intensity of the lines also decreases at the curves in the road. The proposed solution divides the road image into partitions, where the sizes of the partitions are inversely proportional to the distance of the partition to the vehicle.

After the image is partitioned, several preprocessing steps are required before applying the Hough transform. These preprocessing steps should be fast because the Hough transform is already computationally expensive for real time applications. Since edge detection techniques are usually computationally expensive for real time applications [12, 13], each partition is converted to binary images via applying a threshold filter after a color remapping process.

After the image is partitioned, a separate Hough transformation is applied to each single partition. The most intense line in each partition, which is

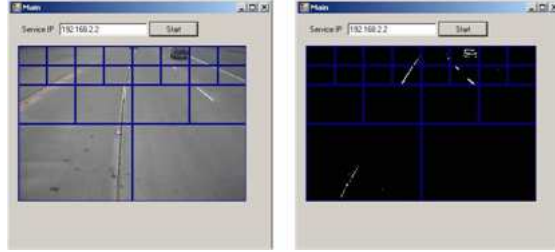


Fig. 1. (a) Partitioned image, (b) Binary image.

the candidate line segment, is taken into consideration in order to find the global lanes in the image. Since the Hough lines are represented in polar coordinates (r, θ) instead of rectangular coordinates (x, y) , the candidate lines are grouped according to their slopes and distances to the center of the image as well as their intensities. The center of the frame is chosen as the bottom point.

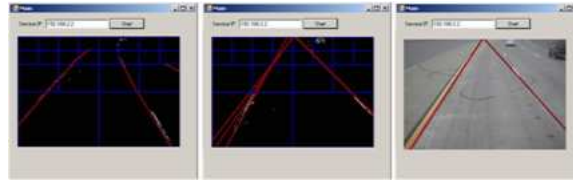


Fig. 2. (a) Candidate lines, (b) Transformed line, (c) Detected lines.

The transformation of the lines basically changes the center point of the polar coordinates for each transformed line which is achieved by the following translation

$$\begin{aligned} r' &= r + (x - x') \cos(\theta) + (y - y') \sin(\theta) \\ \theta' &= \theta \end{aligned} \quad (1)$$

where (r', θ') is the polar coordinates of the transformed Hough line (r, θ) . Note that the translation of the center of the Hough transformation is from (x, y) to (x', y') .

After the lines are grouped, the most intense three clusters are assigned as the lanes. However, there may be less than three lanes if the sum of the intensities of the candidate lines is less than a threshold value.

3.2. Lane Tracking

For lane tracking, HMM is used to represent the relation between the current frame and its successor. Each line in a specific frame is represented by an individual (r, θ) pair. In the succeeding frame, the process will most probably observe the same line at (r', θ') which is not very far from the position of the line in the previous frame. The probability of observing (r', θ') pair in the next frame is modeled as an HMM problem. In addition, θ and r values are modeled by two different HMM. The θ value is discretized as $(0, 1, 2, 3, \dots, 178, 179)$ where the r value is discretized at the pixel level. This discretization schema is used in both transmission and emission matrices. The emission probability matrix shows the probability of observing θ' (or r') in the next frame, having observed θ (or r) in the current frame. In our implementation, the observation and state transition matrix values are derived from two Gaussian distributions with different deviations. The deviation of the transition matrix is assigned to a smaller value than the observation matrix, which means, the state transition matrix aims to preserve the current state where the observation matrix promotes the exploration behavior.

4. Experiment

The proposed approach is implemented and tested on a relatively short video sequence of an urban drive. In addition, the new approach is compared with the classical Hough transform where the entire image is processed and the most intense lines are accepted as candidate lines. The properties of the video are as follows.

Table 1. Properties of video sequence.

Camera Position:	Front console of the car.
Resolution:	512x288
Frame Rate:	29.97
Length:	34 sec.

4.1. Setup

As the first step of the experiment, the image is converted to a binary image by using color remapping. The mapping for each pixel from 24bit RGB value to binary value is given in Table 2.

This binarization favors the white and yellow parts of the images. The

Table 2. Color remapping.

Pixel Value	Red	Green	Blue
0-176	0	0	0
176-196	1	1	0
196-255	1	1	1

values are manually crafted for the video sample. More discussions about improving the color remapping can be found in the next section.

The next step is to determine the partitions of the image on which the Hough transforms will be applied. Although the image is 288 pixels high, only the bottommost 116 pixels are used since the road remains in this lower part of the image. The accuracy of this assumption may slightly differ depending on the slope.

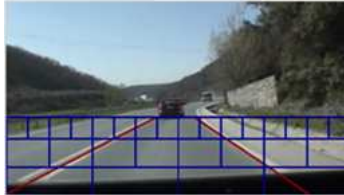


Fig. 3. Image partitions.

The widths of the partitions are 32, 64, and 128 pixels from top to bottom. And the heights are 32, 42, and 42 pixels respectively as shown in Figure 3. These values are assigned according to the position of the camera. After the partitions are calculated, Hough transformation is applied to each partition as described in the previous section. The most promising three lines are assigned as the candidate lane markers. But there may be less than three lines if the intensity of the calculated lines are less than an empirically assigned threshold. The experiment shows that the proposed approach usually detects only two lines most of the time.

After finding the lane markers, HMM method is used to track the lanes. The values of the emission and transition matrices are derived using Gaussian assumption. The deviation of the transition matrix is assigned as 1 and the deviation of the emission matrix is taken as 2. Two separate models are prepared for the θ and r values of the candidate lane markers. The transition and emission matrices are given in Tables 3 and 4. Since the θ values 0 and 179 are actually very close, the emission and transmission values are the same for 1 and 179 in θ matrices. In addition, the range of

the r matrices is (0, 282) because the maximum possible distance for any detected line is 282 pixels where the height of the processed part of the image is 116 and width of the image is 512.

Table 3. (a) Transmission matrix for r , (b) Transmission matrix for θ .

r	0	1	2	...	281	282
0	0.3989	0.2420	0.0540	...	0.0000	0.0000
1	0.2420	0.3989	0.2420	...	0.0000	0.0000
2	0.0540	0.2420	0.3989	...	0.0000	0.0000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
280	0.0000	0.0000	0.0000	...	0.2420	0.0540
281	0.0000	0.0000	0.0000	...	0.3989	0.2420
282	0.0000	0.0000	0.0000	...	0.2420	0.3989

θ	0	1	2	...	178	179
0	0.3989	0.2420	0.0540	...	0.0540	0.2420
1	0.2420	0.3989	0.2420	...	0.0044	0.0540
2	0.0540	0.2420	0.3989	...	0.0001	0.0044
⋮	⋮	⋮	⋮	⋮	⋮	⋮
177	0.0000	0.0000	0.0000	...	0.0000	0.0000
178	0.0000	0.0000	0.0000	...	0.0000	0.0000
179	0.0000	0.0000	0.0000	...	0.0000	0.0000

Table 4. (a) Emission matrix for r , (b) Emission matrix for θ .

r	0	1	2	...	281	282
0	0.1995	0.1760	0.1210	...	0.0000	0.0000
1	0.1760	0.1995	0.1760	...	0.0000	0.0000
2	0.1210	0.1760	0.1995	...	0.0000	0.0000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
281	0.0000	0.0000	0.0000	...	0.1995	0.1760
282	0.0000	0.0000	0.0000	...	0.1760	0.1995

θ	0	1	2	...	178	179
0	0.1995	0.1760	0.1210	...	0.1210	0.1760
1	0.1760	0.1995	0.1760	...	0.0648	0.1210
2	0.1210	0.1760	0.1995	...	0.0270	0.0648
⋮	⋮	⋮	⋮	⋮	⋮	⋮
178	0.0000	0.0000	0.0000	...	0.0000	0.0000
179	0.0000	0.0000	0.0000	...	0.0000	0.0000

4.2. Results

The proposed approach managed to detect and track at least one line in most of the sequence. In addition, false positives are reduced to an acceptable level. In order to validate the results, the proposed approach is compared with the classical Hough Transform approach. In this method, the same part of the image is processed using the Hough transform routine. The most intensive 10 lines are merged according to their r and θ values. Finally three or less candidate lines are selected as the lane markers.

The major differences between the two approaches are shown in Figure 4. The images on the left hand side are the detected or missed lines by the classical approach. The right hand side images are the outputs of the new approach for the same frames which show that the new approach is more robust and accurate.

The computational cost of the proposed approach can be compared as follows. The average processing time is 21.25 milliseconds for a laptop PC with Intel T2050 processor at 1.6 GHz whereas the average cost of the classical approach is 15.29 milliseconds.

The results of the experiments for both the proposed approach and the classical method can be found in the project website [15].

5. Conclusions

Lane tracking is one of the major tasks in autonomous urban driving. A hybrid solution of MHT with HMM is applied in this work, and the performance of the system is increased. However there are certain assumptions and shortcomings of the proposed approach. First of all, variable lightning and road conditions require adaptive color remapping. Although this is beyond the scope of this work, it is crucial for a final product. In addition, the proposed approach models the lane boundaries as lines, therefore an approximation is inevitable at curves. However, it is also possible to use combination of line segments which are detected at each image partition. As another future work, the emission matrix can be updated on the fly by already made decisions.

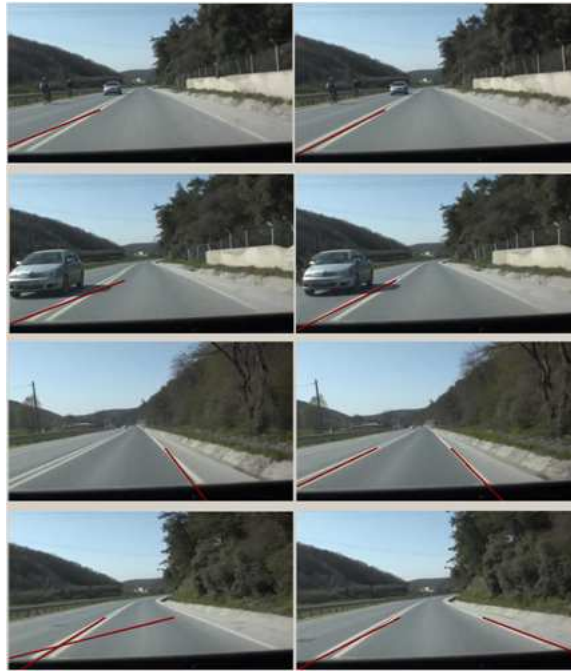


Fig. 4. Differences between classical Hough transformation and proposed approach.

References

- (1) P.V. Hough (1962) Methods and means for recognizing complex patterns. US Patent 3069654.
- (2) Li, Q., Zheng, N. and Cheng, H. (2004) Springrobot: A prototype autonomous vehicle and its algorithms for lane detection. *IEEE Transactions on Intelligent Transportation Systems*, 5(4):300-308.
- (3) Yu, B. and Jain, A.K. (1997) Lane boundary detection using a multiresolution Hough transform.
- (4) McCall, J.C. and Trivedi, M.M. (2006) Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):20-37.
- (5) Dean Pomerleau (1995) Neural Network Vision for Robot Driving. In *The Handbook of Brain Theory and Neural Networks*.
- (6) Kang, D.J. and Jung, M.H. (2003) Road lane segmentation using dynamic programming for active safety vehicles. *Pattern Recognition Letters*, 24(16):3177-3185.
- (7) Wang, Y., Teoh, E.K. and Shen, D. (2004) Lane detection and tracking using B-Snake. *Image and Vision Computing*, 22(4):269-280.
- (8) Kreucher, C., Lakshmanan, S. and Kluge, K. (1998) A driver warning system based on the LOIS lane detection algorithm.
- (9) Apostoloff, N. and Zelinsky, A. (2003) Robust vision based lane tracking using multiple cues and particle filtering.
- (10) Zhou, Y., Xu, R., Hu, X. and Ye, Q. (2006) A robust lane detection and tracking method based on computer vision. *Measurement Science and Technology*, 17(4):736-745.
- (11) McCall, J.C. and Trivedi, M.M. (2006) Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):20-37.
- (12) Canny, J. (1986) A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679-698.
- (13) Ratnayake, K. and Amer, A. (2006) An FPGA-Based Implementation of Spatio-Temporal Object Segmentation. In *Image Processing, 2006 IEEE International Conference on.*, pages 3265-3268.
- (14) Rabiner, L. and Juang, B. (1986) An introduction to hidden Markov models. *iee assp magazine*, 3(1 Part 1):4-16.
- (15) ADES project website, <http://robot.cmpe.boun.edu.tr/ADES>.