ROBUST LANE DETECTION AND TRACKING WITH RANSAC AND KALMAN FILTER

Amol Borkar, Monson H. Hayes III, Mark T. Smith

Address

ABSTRACT

Abstract goes here

***Index Terms*—** One, two, three, four, five

1. INTRODUCTION

Intro

2. PRIOR RESEARCH

Vision based lane detection is a very heavily researched topic. Numerous techniques have been developed over the years with attempts to robustly detect lanes. In the domain of feature extraction, one of the most commonly applied technique is the edge detection based approach [1, 2]. In this method, a Canny edge detector is used to generate the binary edge map. This is followed by computing the classical Hough transform to reflect the ideal orientation of the lane markers on the road. While this approach shows good results in general, it is often skewed by cracks and navigational text on the road. The application of color segmentation to extract lane markers is another strategy [3, 4]. Unfortunately, color segmentation is very sensitive to ambient light and requires additional processing to counter these unpredictable effects. Advancing into image perspectives, majority of the procedures rely on performing feature extraction on the camera perspective image [5, 1, 3, 6]. Although dealing with camera perspective allows access to raw data values, defining of the required features can be rather complicated. The complication arises from dynamics in modeling the sought after features due to their change in shape and size on every row in the image. This type of inconsistency leads to the formulation of unique definitions on every scanline.

Lane detection is a vital component of almost all vision based driver aids. Many of the systems discussed above perform well under a variety of assumptions. Some of the prominent assumptions being the presence of strong lane marker contrast and roads devoid of artifacts like cracks, arrows or similar markers. Unfortunately, these assumptions does hold in many high traffic urban streets and highways. No existing publicly cited literature documents all around performance of a lane detection system on all streets and highways around the world; hence, there is still scope for improvement as robust lane detection still remains unsolved.

3. METHODOLOGY

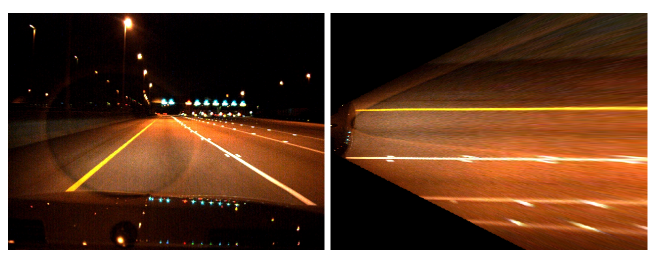
This paper is an extension over the existing layered lane detection approach in [5]. Each module is detailed below.

3.1. Image Enhancement

The acquired Bayer array images are reconstructed into their RGB representation and then converted to grayscale. This is followed by a temporal blurring process which helps to connect the dashed lane marker sequence in the image serving as an enhancement stage [5].

3.2. Inverse Perspective Mapping

Inverse perspective mapping (IPM) is an image transformation technique used to remove the perspective effect from an image. As seen in Fig. 1, application of this transform changes the appearance of the image from a forward facing camera view to a birds eye view. [7, 8, 9]. The benefit



(a) Camera perspective view (b) Birdseye view

Fig. 1: Inverse perspective mapping transforms a camera perspective image into a birdseye view image

of this approach is the simplification of lane marker detection and classification as the initially converging lane marker sequences now appear parallel. In addition, the transformation enables a mapping between pixels in the image plane to their corresponding locations in the world with metric coordinates. Camera parameters such as height from the ground, inclination and horizontal/vertical viewing angles need to be determined ahead of time to guarantee an accurate transformation. The IPM transform is applied to the average image acquired after application of the enhancements above.

3.3. Lane Candidate Location Detection

An adaptive threshold is applied to the IPM average image to generate a binary image which is split into two halves [5]. A low resolution Hough transform is then computed on the binary images and a set of X highest scoring lines are formulated into a list for each half image [5]. Each line in the list is then sampled along its length at specific distance co-ordinates as symbolized by the red "+" in Fig. 2. The corresponding location of each of these points in the IPM average image is recovered. A 1-D search window centered at each

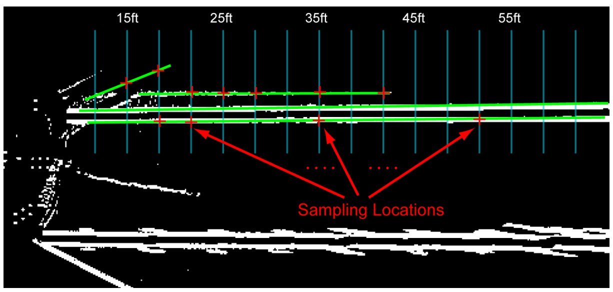


Fig. 2: Distances where the detected lines will be sampled in each half image. Cyan lines are the specific distance co-ordinates. Green lines are the Hough transforms' X highest scoring lines detected in the binary image. +'s are a few locations where the X lines are to be sampled.

point in the IPM average image is used in the matched filter. With the birds eye view, the visualization of the lane marker sequence not only appears parallel, but each lane marker also maintains a constant width in the entire image; consequently, a fixed size and fixed variance Gaussian kernel is used in the matched filter. This was not the case in [5] where a variety of variances had to be used to create the kernel on different scanlines. Matched filtering is iteratively performed on the remaining X-1 lines. As each of the X lines is sampled at the specific co-ordinates, X filtering results are available at each of these positions upon completing the iterations. As a result, the location with the highest correlation coefficient at each distance co-ordinate is chosen as the best estimate as shown in Fig. 3.

3.4. Outlier Elimination and Data Modeling

Upon acquiring the collection of best estimates, Random Sample Consensus (RANSAC) is applied to the data points. The generic RANSAC algorithm robustly fits a model through the most probable data set or inliers while rejecting outliers [10, 11]. Linear Least Squares Estimation (LSE) is then used to fit to a line on the inliers. The orientation of line

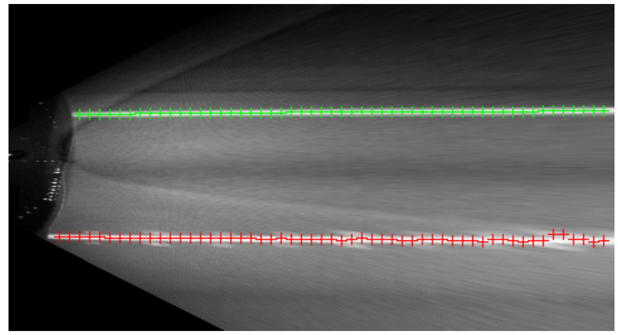


Fig. 3: The best estimates for lane marker candidates in each half image

is modeled in terms of ρ and θ with respect to the origin (top left of the image). For data fitting, a line was chosen over a parabolic fit since the latter is more sensitive to minor perturbations within the inliers, this can result sometimes in desirably shaped curves.

3.5. Tracking

The orientation of the fitted line is predicted using a Kalman filter. The state vector $x(n)$ and observation vector $y(n)$ are defined as

$$x(n) = y(n) = \begin{bmatrix} \rho \\ \dot{\rho} \\ \theta \\ \dot{\theta} \end{bmatrix} \tag{1}$$

where ρ and θ are parameters used to model the orientation of the line, while $\dot{\rho}$ and $\dot{\theta}$ represent the derivatives of ρ and θ computed over the current and previous frames. Piece-wise linearity is assumed between the frames allowing use of the Kalman filter [12, 13]. The state transition matrix A and observation model C are define as

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

The independence between the variables in $x(n)$ and $y(n)$ allows creation of simple covariance matrices Q_w and Q_v. Q_w and Q_v represent the process and observation noise respectively [12]. The covariance matrices are defined as identity and multiplied with non-uniform weights along the diagonal, these weights correspond to the variances of the parameters in $x(n)$ and $y(n)$. The Kalman filter recursively predicts the parameters in the state vector from the previously available information [12, 13].

In the case of a lane marker sequence not being detected, the values in Q_v are increased significantly and $\hat{x}(n|n)$ is modified as Eq. 4

$$\hat{x}(n|n) = \hat{x}(n|n-1) \tag{4}$$

to force the Kalman filter to rely purely on prediction.

Finally, after extraction from $\hat{x}(n|n)$, ρ and θ are transformed back to the image plane and used to model the estimated orientation of the lane marker sequence. The series of iterated matched filtering and tracking is similarly performed on the other half image.

4. EXPERIMENTAL ANALYSIS

4.1. Hardware

The test bed is equipped with an Intel based computer and necessary hardware to provide power. A forward facing Firewire color camera installed below the rear-view mirror has a clear view of the road ahead. The Firewire specification allows capture at VGA resolution and 30fps. Video from the camera is recorded onto hard disks as a sequence of still images.

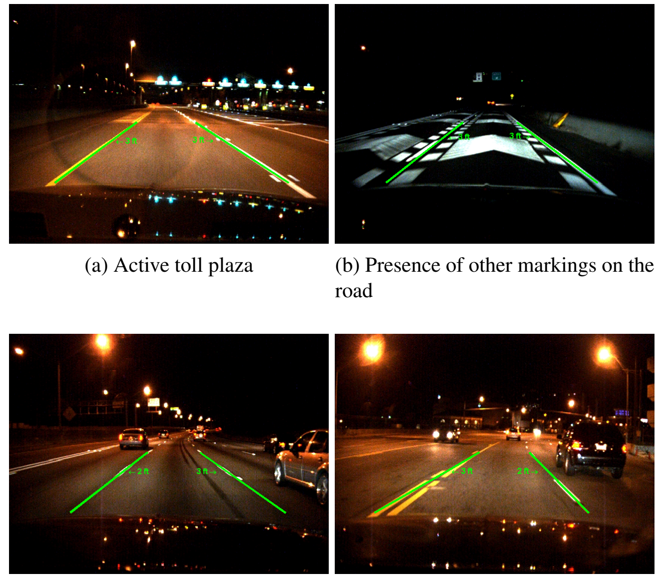
4.2. Results

Algorithms required for lane detection were designed in Matlab and require a computational time of approximately 0.8 seconds per frame. The results in Table 1 illustrate the performance of the proposed system when tested on over 10 hours of captured video. The results also show improvement in accuracy over the system developed in [5].

Table 1: Comparing accuracies of lane detection system

Road type	Traffic type	Average Accuracy Per Minute					
		Current System			Previous System [5]		
		Correct	Incorrect	Misses	Correct	Incorrect	Misses
Isolated Highway	Light	0%	0%	0%	0%	0%	0%
	Moderate	0%	0%	0%	0%	0%	0%
Metro Highway	Light	0%	0%	0%	0%	0%	0%
	Moderate	0%	0%	0%	0%	0%	0%
City Streets	Variable	0%	0%	0%	0%	0%	0%

The captured videos contain scenes with a variety of traffic and illumination conditions that depict environments a real-world system would encounter [5]. Fig. 4 shows a few of instances of accurately detected lane marker sequences. The metric used to test the quality of the lane detections is accuracy per minute. This metric allows portability and consistency when testing videos with different frame rates. Since defining a ground truth for the data is extremely tedious, it is commonly avoided. As a result, computing the error between ground truth and prediction is not possible. Detections are



(a) Active toll plaza (b) Presence of other markings on the road

(c) Busy Highway (d) Busy city streets

Fig. 4: Successful lane detections in various environments

generally quantified based on visual inspection. Fig. 5 shows a comparison between the observed and predicted value of ρ. As pure measurements tends to be noisy, the Kalman filter in

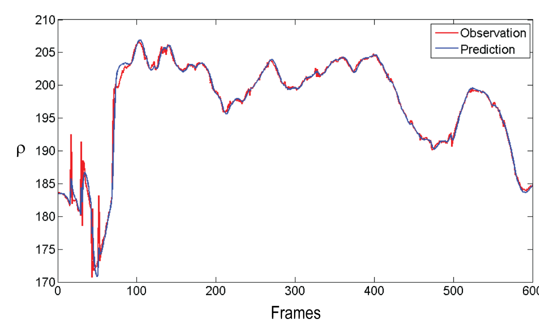
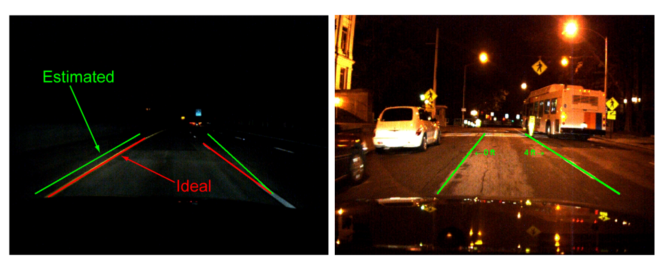


Fig. 5: Comparison between observed and predicted values of ρ over a range of frames

this context serves as a low-pass filter by smoothing the observed values.



(a) Unsettled after a bump (b) Poorly maintained city streets

Fig. 6: Inaccurate lane detections in a few scenes

A few instances of incorrect lane detections are shown in Fig. 6. Fortunately in Fig. 6a, the Kalman filter is able to settle within a couple of milliseconds after passing the bump on the road. However, in Fig. 6b, the absence of lane markers due to aging and wears leads to detection and tracking false positives like cracks.

5. CONCLUSIONS

The work presented in this paper is a significant improvement over the layered lane detection system presented in [5]. The addition of features such as i) Inverse Perspective Mapping (IPM) ii) Random Sample Consensus (RANSAC), and iii) Kalman filtering has added to the novelty and extension over the previous system. IPM aided in simplifying the process of finding candidate lane markers, while RANSAC helped in rejecting outliers within the estimations. Finally, the Kalman filter ignored minor perturbations and kept the lane marker sequence on its track.

The data set used to test the accuracy of the proposed system was recorded on Interstate highways and city streets in and around Atlanta, GA. Despite the variety in traffic conditions and road quality encountered, the proposed system still yielded good performance as reflected in Table 1.

6. FUTURE WORK

Lane Departure Warning (LDW) will be implemented in the not too distant future as the proposed lane detection system is able to accurately determine the distance to the lane markers on either side. By analyzing the velocity and acceleration of lane marker movement, the driver can be notified of an up coming lane change. Further investigation is need to enable day time lane detection as current assumptions lead to less than satisfactory results. In addition, the implemented algorithms will be ported to C++ with the help of existing libraries like OpenCV and VXL to facilitate a real-time system.

7. REFERENCES

[1] A.A.M. Assidiq, O.O. Khalifa, R. Islam, and S. Khan, "Real time lane detection for autonomous vehicles," in *International Conference on Computer and Communication Engineering*, 2008, pp. 82–88.

[2] C. C. Wang, S. S. Huang, and L. C. Fu, "Driver assistance system for lane detection and vehicle recognition with night vision," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, 2005, pp. 3530–3535.

[3] T. Y. Sun, S. J. Tsai, and V. Chan, "HSI color model based Lane-Marking detection," in *IEEE Intelligent Transportation Systems Conference*, 2006, pp. 1168–1172.

[4] K. Y. Chin and S. F. Lin, "Lane detection using color-based segmentation," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2005, pp. 706–711.

[5] A. Borkar, M. H. Hayes, M. T. Smith, and S. Pankanti, "A layered approach to robust lane detection at night," in *2009 IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems*, 2009.

[6] H. Wang and Q. Chen, "Real-time lane detection in various conditions and night cases," in *Proceedings of the IEEE Intelligent Transportation Systems Conference*, 2006, pp. 17–20.

[7] S. Sehestedt, S. Kodagoda, A. Alempijevic, and G. Dissanayake, "Robust lane detection in urban environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 123–128.

[8] Y. Shu and Z. Tan, "Vision based lane detection in autonomous vehicle," in *Fifth World Congress on Intelligent Control and Automation*, 2004, vol. 6, pp. 5258–5260.

[9] M Bertozzi and A Broggi, "GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 62–81, 1998.

[10] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2 edition, Apr. 2004.

[11] Martin A. Fischler and Robert C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[12] Monson H. Hayes, *Statistical Digital Signal Processing and Modeling*, Wiley, Mar. 1996.

[13] Eli Brookner, *Tracking and Kalman Filtering Made Easy*, Wiley-Interscience, Apr. 1998.