Lane Detection and Tracking Using a Linear Parabolic Model

Amin Boroun

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
February 2015
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Serhan Çiftçioğlu
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Hasan Demirel
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Erhan A. İnce
Supervisor

Examining Committee

1. Prof. Dr. Hasan Demirel
2. Prof. Dr. Hüseyin Özkaramanlı
3. Assoc. Prof. Dr. Erhan A. İnce

ABSTRACT

All over the world, safety is an important issue for humans. Therefore, reducing road accidents and hence, saving lives of individuals has become a great research interest in the context of advanced driver assistance systems. Among the whole complex and challenging tasks of future road vehicles, road lane detection or road boundaries detection will hold a key role. Lane detection and tracking is an important topic in autonomous navigation, since the navigable region usually stands between the lanes, especially in urban environments. Several approaches have been proposed for lane detection, but Hough transform seems to be the dominant among all and in the current thesis it is employed for lane detection aims. A robust lane tracking method is also required for reducing the effects of the noise and achieving the demanded processing time. Herein, a lane tracking method using linear parabolic model as a practical method has been applied for following the lanes. Next to the detection process, detected boundaries in consecutive video frames are tracked through a linear- parabolic model. The linear part of the model is used to fit the near vision field, while the parabolic model fits the far field. Connected component method was utilized to improve the lane detection process which was of great importance for the curved part of the road. The data from Carnegie Mellon University directory were used for applying the hybrid method and then three of them namely run2a, run2b and May30-90 were simulated. Thesis also provides simulation results for two custom videos from FAMAGUSTA-NICOSIA road taken under different lighting conditions. These sequences are named ROAD-22 and ROAD-23in this study. The achieved results demonstrated the outstanding performance of the proposed hybrid

method. For all the available samples, the average percentage values for correctly detecting and marking the right and left lines were 96.3% and 97.12% respectively.

Keywords: lane detection; lane tracking; Hough transform; linear-parabolic model; connected component analysis.

ÖZ

Güvenlik konusu, dünya çapında insanoğlu için en önemli konulardan biri konumundadır. Dolaysıyla trafik kazalarının azaltılması ve sonuç olarak insan hayatının korunması, Gelişmiş Sürücü Yardım Sistemleri kapsamında önemli bir araştırma konusu haline gelmiştir. Yol şeridi veya yol sınırlarını bulma konusu, gelecek nesil araçların tüm karmaşık ve zorlu görevlerinin arasında kilit bir rol oynamaktadır. Özellikle kentsel çevrelerde olmak üzere, gezinecek olan alanın genellikle iki şerit arasında bulunmasından dolayı, şerit bulma ve izleme konusu özerk navigasyonda önemli bir konuyu oluşturmaktadır. Şerit bulma konusunda farklı yaklaşımlar önerilmiş olmakla birlikte, Hough dönüşümü tüm yaklaşımlar arasında baskın bir role sahip olup mevcut tez çalışmasında da şerit bulma amacı için bu yaklaşımdan yararlanılmıştır. Ayrıca parazit etkilerinin azaltılması ve arzu edilen işlem sürelerine ulaşılması için güçlü bir şerit izleme yöntemine gereksinim duyulmaktadır. Burada şeritlerin izlenmesi için pratik bir yöntem olarak çizgisel parabolik modeli kullanan bir şerit izleme yönteminden yararlanılmıştır. Şerit bulma işlemini takiben, ardışık video karelerinde bulunan sınırlar, çizgisel-parabolik bir model sayesinde izlenmektedir. Modelin parabolik bölümü uzak görüş alanlarının uyarlanmasında kullanılır iken, modelin çizgisel bölümü ise yakın görüş alanının uyarlanması için kullanılmaktadır. Bağlantılı bileşen yöntemi, yolun kavisli bölümü için büyük önem taşıyan şerit bulma sürecinin geliştirilmesi için kullanılmıştır. Karışık yöntemin uygulanması için Carnegie Üniversitesi veri tabanındaki veriler kullanılmış olup daha sonra ise run2a, run2b ve May30-90 isimli veriler benzeştirilmiştir. En son veri tabanı ile birlikte farklı aydınlık düzeylerine sahip FAMAGUSTA-NICOSIA yolundan seçilen ROAD-22 ve ROAD-23 isimli iki video

kaydedilmiştir. Elde edilen sonuçlar, önerilmiş olan karışık yöntemin iyi bir performans verebileceğini kanıtlamıştır. Mevcut tüm örnekler (çerçeveler) kullanıldığında sağ ve sol şeritleri doğru olarak belirleyip işaretleme oranları % 96.3 ve % 97.12 olarak belirlenmiştir.

Anahtar Kelimeler : Şerit bulma; şerit izleme; Hough dönüşümü; çizgisel-parabolik model; bağlantılı bileşen analizi.

ACKNOWLEDGEMENT

Foremost, I would like to express my sincere gratitude to my supervisor Assoc. Prof. Dr. Erhan A. İnce for his patience, enthusiasm, motivation and continuous support on my MS research.

I also would like to thanks Dr. Kiyan Parham. Without his kind support I would have found it very difficult and probably not reach the final stage in my thesis work.

Lastly, I would like to thank my parents for giving birth to me at the first place and supporting me spiritually throughout my life.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

Safety is one of the most significant concerning disputes of human being. Regarding this, one of the minor expectations of the people is to reach their destination safely, deprived of any incidents during the travel. Vehicle crashes remain the leading cause of accidental death and injuries in most traffic congested countries e.g. UK, USA, and Asian countries claiming tens of thousands of lives and injuring millions of people each year [1]. Most of these transportation deaths and injuries occur on the nation's highways. The probability of road accidents can be reduced significantly, by getting benefit of improved driving assists.

Therefore, a system that provides a means of warning the driver to the danger, has the potential to save a considerable number of lives. In order to increase safety and reducing road accidents, people are spending lots of money for the advancement in the driving techniques which ensures the safety. The technology makes man to think more to improve the safety to save the lives. The automobiles are more conscious of providing safety feathers like seat belts, air bags and strong body structures which provide the passive safety that may reduce the effects of an accident. Avoiding accidents and saving lives are one of great interests that all researchers and automobile companies work on. In Advanced Driver Assistance Systems (ADSA) in order to achieve the desired safety on roads, the complex and challenging tasks of future road vehicles are road lanes detection or boundaries detection which is exposed in white and black lines on roads. In fact, several researchers worldwide have been developing vision-based systems for lane detection, lane tracking and lane departure warning. However, most of them present limitations in situations involving shadows, varying illumination conditions, bad conditions of road paintings and other image artifacts. In our research we have developed a linear parabolic model to improve the robustness of lane detection and tracking for intelligent transportation systems. In proposed method, lane detection and tracking will be inspected by linear parabolic model, and connected component function for the aim of improving the performance of the lane detection and tracking.

1.1 Lane Detection Overview

Lane detection is one of the methods which use the principle of vision based lane detection. As the name itself indicates is a process of detecting as well as recognizing the lanes where the ground traffic circulates. For driving, Advanced driving assistances of the lane detection is one of the essential functions. The lane detection has become very specific term that implies the utilization of certain perceptive sensors, certain processing units, and certain algorithms to perform this functionality.

The lane detection is processes which have to be effective with the following. There are many factors which affects the lane detection. The good quality of lane should not be affected by shadows of which can be caused by appearances of trees, buildings and other aid boards, the existences of surrounding object, the change of light condition, the dirt left on the road surface etc.

In line detection, in the case of lane marks, some major problems are still unsolved. Detection not only should not assume the roads as straight, but also the curves of the road would be considered by it.

Balancing the image which detects the lane should assume the parallelism of both sides of the lane marking for the aim of improving the detection besides of the noises in images. Despite of several researches done on lane detection, there are lots of difficulties in lane detection. So far, there is not a comprehensive technique which is capable of detecting lanes successfully [2]. In the current thesis, all the above stated concerns about lane detection and tracking have been considered.

1.2 Related Work on Lane Detection

In the following section a comprehensive review of the lane detection and tracking from the literature is done.

Schneiderman and Nashman [3] described a visual processing algorithm that supported autonomous road following. There were three stages of computation: extracting edges; matching extracted edge points with a geometric model of the road, and updating the geometric road model. All processing was confined to the 2-D image plane. No information about the motion of the vehicle was used. The algorithm performed accurately.

Litkouhi, Lee and Craig [4] developed a theory for designing a lane estimator and a lane controller. The roadway curvature and the relative positioning of the vehicle within its lane were estimated using Kalman filtering. Inputs to the estimator were vehicle kinematical variables provided by a vehicle directional control model, and lane boundary information provided by a video camera model. Although the developed model used as input lane information, its detection was not discussed in this particular paper.

Taylor et al [5] lane extraction system was based on a parameterized model for the appearance of the lanes in the images. This model captured the position, orientation and width of the lane as well as the height and inclination of the stereo rig with respect to the road. Their work differed from ours in the premise that they had stereo vision, while here only information from one camera is available.

Betke, Haritaoglu and Davis [6] analyzed color videos taken from a car driving on a highway. The system used a combination of color, edge, and motion information to recognize and track the road boundaries, lane markings and other vehicles on the road. The system recognized and tracks road boundaries and lane markings using a recursive least squares filter. The algorithm here presented could not be adapted to our situation since in relies on color information, while the video here processed is in gray scale.

In 2004, Jung and Kelber [7] addressed the problem of lane detection and lane tracking. A linear model was used to approximate lane boundaries in the first frame of a video sequence, using a combination of the edge distribution function and the Hough transform. A linear-parabolic model is used in the subsequent frames. The linear part of the model is used to fit the near vision field, while the parabolic model fits the far field. The proposed line detection procedure is applied independently to each lane boundary. In their work, information of the dependencies between the lines is used to improve the detection results.

Fletcher, Petersson and Zelinsky [8] develop and evaluate a road scene monotony detector. Again, although the method uses information about lanes, its detection is not discussed in this work.

Hsieh et al. [9] presented an automatic traffic surveillance system to estimate important traffic parameters from video sequences using only one camera. An automatic scheme to detect all possible lane dividing lines by analyzing vehicles' trajectories is proposed.

Maire and Rakotonirainy [10] described a system that analyses videos of driving sessions collected by on-board Web-cameras. The system detects and tracks lane markings in order to estimate the relative position of the vehicle with respect to its lane. The analysis of the video recording is performed in reverse temporal order. Although having several benefits when compared to forward analysis, it makes it not suitable for an on-line system.

McCall and Trivedi [11] developed the "video-based lane estimation and tracking" (VIOLET) system. The system is designed using steerable filters for lane-marking detection. Unlike the present work, several sensors, like front camera, vehicle speed, vehicle steering and vehicle and road model, are used as input.

Isa [12] used image processing to perform some experimental studies on dynamics performance of lateral and longitudinal control for autonomous vehicle. They presented an algorithm of vehicle lane detection and tracking based on color cue segmentation, canny edge detection and Hough transform.

Borkar, Hayes and Smith [13] also describe a lane detection system. The camera captured image undergoes pre-processing in the form of temporal blurring and gray scale conversion. Then, Inverse Perspective Mapping is applied to remove perspective and transform the image into a bird's-eye view. An adaptive threshold converts the gray scale image into binary and then a low-resolution Hough transform is computed to find a set of candidate lane markers. The candidate markers are further scrutinized in a matched filtering stage to extract the lane marker centers. Random Sample Consensus is used to estimate parameters for fitting a mathematical model through the recovered lane markers. Finally, the Kalman filter predicts the parameters of each lane marker line from one frame to the next.

Cheng and Chiang [14] developed an automatic lane following navigation system for the intelligent robotic wheelchair. The system was developed to work in a barrier-free environment and used video paint line detection as the basis of automatic tracking navigation. It is clear that these conditions do not hold in our application.

More recently, a video-based lane detection using a fast vanishing point estimation method was proposed by Benligiray, Topal and Akinlar [15]. The first step of the algorithm is to extract and validate the line segments from the image. In the next step, an angle based elimination of line segments is done according to the perspective characteristics of lane markings. Remaining line segments are extrapolated and superimposed to detect the image location where majority of the linear edge features converge. The location found by this operation is assumed to be the vanishing point. Subsequently, an orientation-based removal is done by eliminating the line segments whose extensions do not intersect the vanishing point. The final step is clustering the remaining line segments such that each cluster represents a lane marking or a boundary of the road. The properties of the line segments that constitute the clusters are fused to represent each cluster with a single line.

Finally, Gopalan et al. [16] used a learning approach towards detection and tracking of lane markings. They proposed the following: 1) a pixel-hierarchy feature descriptor to model the contextual information shared by lane markings with the surrounding road region; 2) a robust boosting algorithm to select relevant contextual features for detecting lane markings; and 3) particle filters to track the lane markings. At the core of the approach is the importance placed on the quality of data. There can be instances such as foggy or rainy road conditions where the visual inputs alone are insufficient to detect lane markings.

The GOLD system developed by Broggi, it used an edge-based lane boundary detection algorithm [17]. Hardware and software architecture based on stereo vision for use on moving vehicles to improve road safety. Based on a full-custom massively parallel hardware, detect generic obstacles and the lane position in a structured environment (with painted lane markings) at a rate of 10 Hz.

Kreucher C. [18] proposed in LOIS the algorithm (Likelihood of Image Shape) has been shown to find robust markings, even in the presence of observation of occlusion and a plurality of light conditions. He uses the algorithm to follow the laws of the road through a sequence of images, and a warning of a crossing is Imminent.

Mellon University developed a system called AURORA [18], the lane marks observed on structured roads as highways and city streets. The lateral position of the vehicle calculated from the detected line marker. If the car begins to stray from the path, alerts the driver with audible and visual alarms AURORA.

Real time vision-based lane detection method is presented to find the position and type of lanes in each video frame [19] proposed a method for lane detection effective combination of filter functions edge-Link channels. The first filter means candidates are sought in the region of interest (ROI). During the research, a broad edge linking algorithm circuit slot marginal land used to produce the filter width for wider access board and serves as a way to research the edge orientation and tape are used to filter the channels marked border pair link candidates. A linear model based method has been developed for detecting the tracking markers in real time. The estimation of the linear model is robust filtering capabilities such as efficient roads and edges, color, width and direction are combined to follow the markings on the parameters of the linear model. Lane position can be determined from the linear model parameters and Lane Departure can be calculated. A diagram of the overall management of the Lane Departure proposed method of detection is shown in Figure 1.1.

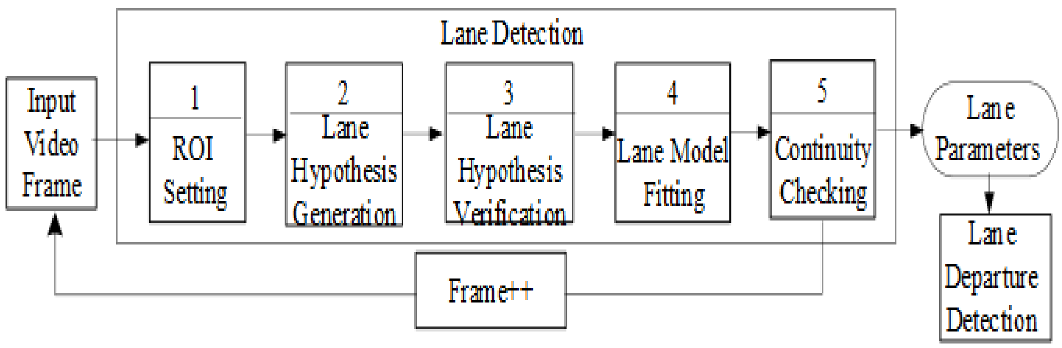


Figure 1.1: Lane detection block diagram [20].

Image characteristics of the lane that comes with a new method for lane detection and tracking can accurately extracted by contrast gray image and processing binary[21]. The filter strengthening increasingly application track binary information. Smooth Gaussian image Canny operator, processed for the detection of channel outlines, when the corner detection method is used for the image coordinates of the corners, finally RANSAC is used to get the optimized lane step by step, the lane parameters used to obtain more accurate track of and extraction of the curve is more perfect. The method not only improves the accuracy of path discovery, but also ensures the safety of the vehicle.

TFALDA is a lane detection algorithm proposed by yam et al [22] TFALDA which stands for Three-Features based Automatic Lane Detection Algorithm presented, suitable for rapid automated detection of lane boundaries in different environments without tedious manual initialization or prior information on the way. The strength of the algorithm is to merge all three of the main characteristics of a lane boundary, and control of the temporal evolution. TFALDA could develop into a solid wide variety road conditions by optimizing the parameters of an evolutionary algorithm, instead of manually testing and improving. The lanes can be detected inverse perspective transformation in an image plane view from the steering angle next road and the information needed to control the direction of the channel's output can be obtained directly.

Chapter 2

LANE DETECTION BASED ON HOUGH TRANSFORM

2.1 Lane Detection Background

Lane detection is a well-researched area of computer vision with applications to autonomous vehicles and driver assistance systems. This is partly because, despite the apparent simplicity of the white markings on a dark road, making it very difficult to identify the markings on different types of roads. These difficulties are of an occlusion in the shadow of other vehicles, changes in the roadway itself, and different types of road markings. A lane detection system must collect all types of markers roads confusion and filtered to give a reliable estimate of the path of the vehicle's position. Lane detection plays an important role in driver assistance systems. In general, the steps of lane detection localize lane boundaries in the images of the specified path, and can help to estimate the geometry of the floor and lateral position ego vehicle on the road, Lane detection in intelligent cruise control environments for Lane Departure Warning, modeling the way, and so on.

2.2 Lane Detection Algorithm (LDA) Overview

Lane detection algorithms detect lane markings and the edges of the road, and estimate the vehicle position in the lane. Lane detection provides a framework for the support of many other single-camera based Mobil eye functions as vehicle detection. In this case, it contributes to the correct position of the vehicle in the same lane. Provided that the road markings visible and that their testimony is not hindered by the presence of clutter like acknowledge shadows, rain, snow or other disturbances on the road. The LDA recognizes the majority of white, blue and yellow markings across the world, and Mobil eye system is approximately 99% of cases.

Different types of marks, such as solid, dashed, Bot points are double and triple road markings validated and integrated into production successfully. In addition, recognizing the LDA roadside (road edges) unmarked, such as grass or gravel banks, for more information on the adjacent track to support the strategy of caution and refine the OEM requirements. Also developed a system of permits for better separation of ambiguous markings, road markings double, triple, markings, etc. The system has been refined and adapted to meet the variation found in different countries correctly. The authorization mechanism can also use the color information for better separation.

The LDA was tested in a series production programs in Europe, North America, Africa, the Middle East and Asia and has been validated on several continents and in a wide range of scenarios, including bright sunlight and weather around the world. In construction areas where there are many overlapping brands, the system is not available. Lane markers of different colors (e.g. blue markings Korean) has successfully developed and operated on the same input a monochrome imager as all other functions.

Mobil eye currently working on a back-LDA with existing units (rear-facing cameras) already in production for recycling applications. This increases the LDW function in difficult situations, such as when entering a tunnel or drivers support in situations where, for example, the sun blinds front camera, and there are reflectors on road seams and tar caused the overall system performance is improved.

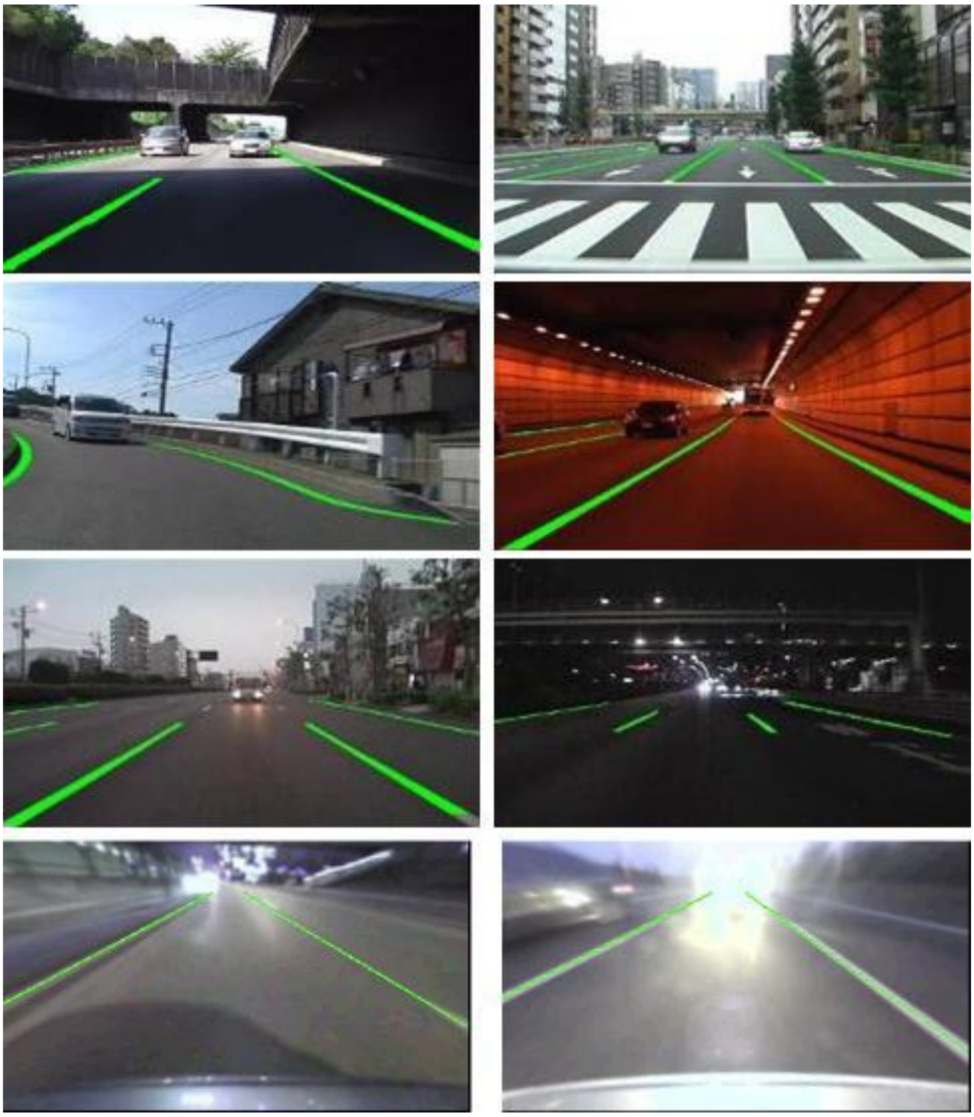


Figure 2.1: Example of lane detection [20].

2.2.1 Features of Lane Marking

1) Adapts to various types of roads.
2) Color, style and width of markings recognition.
3) Detects all road markings in the picture.
4) Integrated navigation system, see the track ego lane change and offer advice.
5) Adapts to different weather and light condition.

2.2.2 Types of Road Lines

1) **Continuous center lines**

You can cross a continue center line to enter or leave a road, but cannot overtake.

2) **Broken Center Lines**

You are allowed to overtake across a broken Centre line or broken Centre line.

3) **Continuous Edges Lines**

Boundary lines (edges lines) are used to select the edge of the road. The area to the left edge of the line is the axis of the road which is also called shoulder of the road. This is not just an extra lane for vehicles to travel in, but cyclists may also travel on the shoulder road. Vehicle also used the road edges lines in case when vehicle entering or leaving the road, stopping at the side of a road, turning at an intersection etc.

2.3 Lane Detection Using Hough Transform

In this section we will explain the lane detection by using gray scale method and edge detection and will examine the results by using this method in chapter 4.

2.3.1 Defining and Conversion of RGB into Gray Scale

In this step, the captured color image is converted to gray scale to make method faster, less computational, and less sensitive to scene condition[23]. In our proposed method, captured images choose from directory of Carnegie Mellon University named run2a, would be processed. The camera is adjusted in a way that the vanishing point of road should be placed on the top of Region of interest as shown in Figure 2.2 based on camera place adjustment.

Figure 2.2: Gray scale of RGB image.

2.3.2 Canny Edge Detection

By applying the optimum local thresholding to selected part of image, ROI, we have binary image as the input for this step. In this step, to find lane boundaries in the image we use one of the edge detection methods called Canny Edge Detection and the detected boundaries are shown in fig 2.3.

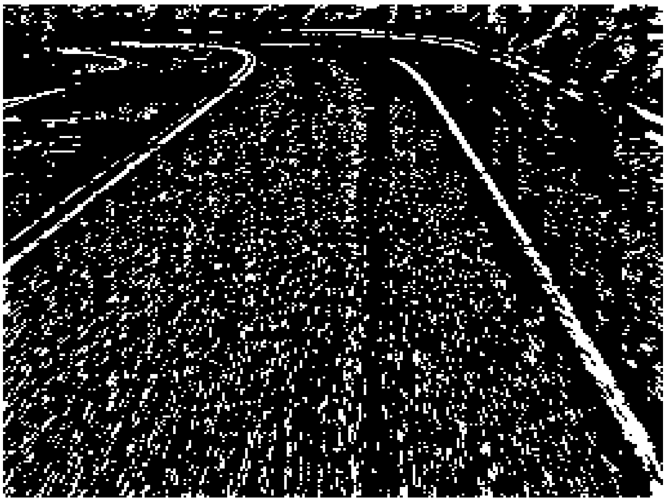


Figure 2.3: canny edge detection.

Canny Edge detector most commonly used for step edges due to optional then is corrupted by white noise. The objective is the detected edges that must be as close as possible to the true edges. The number of local maxima around the true edge should be minimum.

Canny edge detection basically uses gradient vector of an intensity image. Lane boundaries have high contrast in the image, and this feature yields high values of gradient vector by which we can find the edge direction, which is orthogonal to gradient vector. Many edge detection methods are based on this principle, but the efficiency levels are different. One of the best and efficient methods is canny edge detection. The most important characteristics of canny method are that the error rate of this method is low because this algorithm uses double thresholding. Therefore, the detected edge is really close to true place. We should also mention that canny edge detector is very sensitive to the noise.

2.4 Hough Transform Line

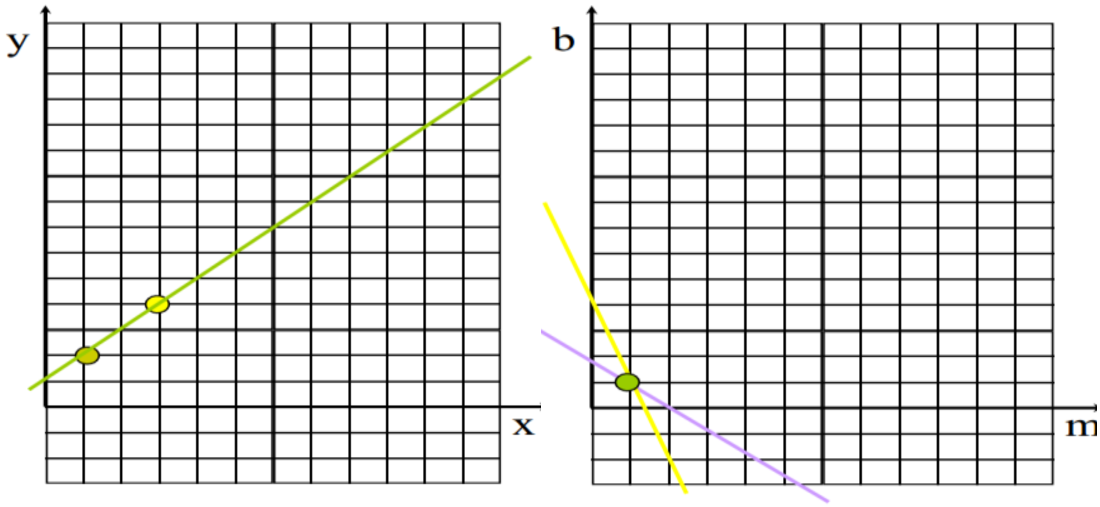
The Hough transform is an image processing technique for feature extraction developed by Paul Hough in 1962. It is more commonly used for detection of lines in an image, but can also be used to detect any arbitrary shapes, for example circles, ellipses, and so on. For this project, it was used for its more common purpose. The underlying principle of the Hough transform is that every point in the image has an infinite number of lines passing through it, each at a different angle. The purpose of the transform is to identify the lines that pass through the most points in the image, i.e. the lines that most closely match the features in the image. To do this, a representation of the line is needed that will allow meaningful comparison in this perspective. A second line is drawn from the origin to the nearest point on the line at right angles. The angle that this second line makes to the origin is recorded, as is the distance from the origin to the point where the two perpendicular lines meet.

2.4.1 Finding Straight Lines

Consider a pixel in position (X_k, Y_k) equation of a straight line,

$$Y_K = m X_k + b \tag{2.1}$$

Set b=- $m(X_k, Y_k)$ and draw this (single) line in ”*mb*-space” Consider the next pixel with position (X_j, Y_j) and draw the line b=- $m(X_j + Y_j)$ ”*mb*-space” (also called parameter space). The points (m’, b’) where the two lines intersect represent the line $y=m'x+b'$ in”*xy*-space” which will go through both (X_k, Y_k) and (X_j, Y_j). Draw the line in *mb*-space corresponding to each pixel in *XY*-space. Divide *mb*-space into accumulator cells and find most common (m’, b’) which will give the line connecting the largest number of pixels.

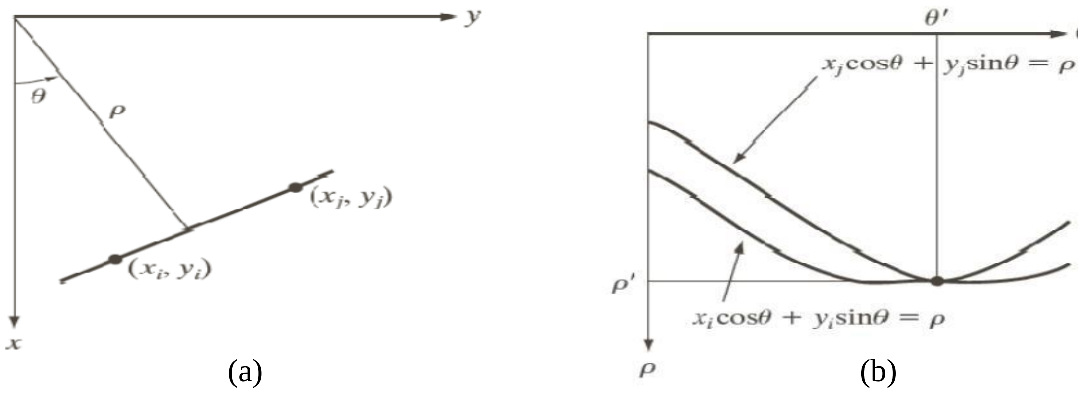


(a) (b)

Figure 2.4: Hough Transform in the *x-y* and *m-b* spaces
(a) Hough Transform *x-y* space, (b) Hough Transform *m-b* space[20]

In reality we have a problem with y=mx +b because m reaches infinity for vertical lines, so we use:

$$X\ cos\ \theta + y\ sin\ \theta = \rho \tag{2.2}$$



(a) (b)

Figure 2.5: Parameterization of a line in *x-y* plane and a sinusoidal curve in the ρ-θ plane
(a) parameterization of a line (b) parameterization of a sinusoidal curve

By using Hough Transform for a too big cell and we merge quite different lines too small and noise causes lines to be missed ,count the peaks in the Hough array, treat adjacent peaks as a single peak ,search for points close to the line and iterate the procedure.

Detecting shapes or features in a digital image is important for some purposes like detection of straight lines. In order to find lines in an image, we use standard Hough line transform that is one form of Hough Transform. For detecting the lines, we consider the output of Edge detection step as the input of Hough line detection, and this transformation finds lines in an image based on figure 2.4,2.5, describes that every point in Hough space is a line in Euclidean space and vice versa. By using this basic we detect the lines in an image obtained from edge detection step, and it is shown in Figure 2.8.

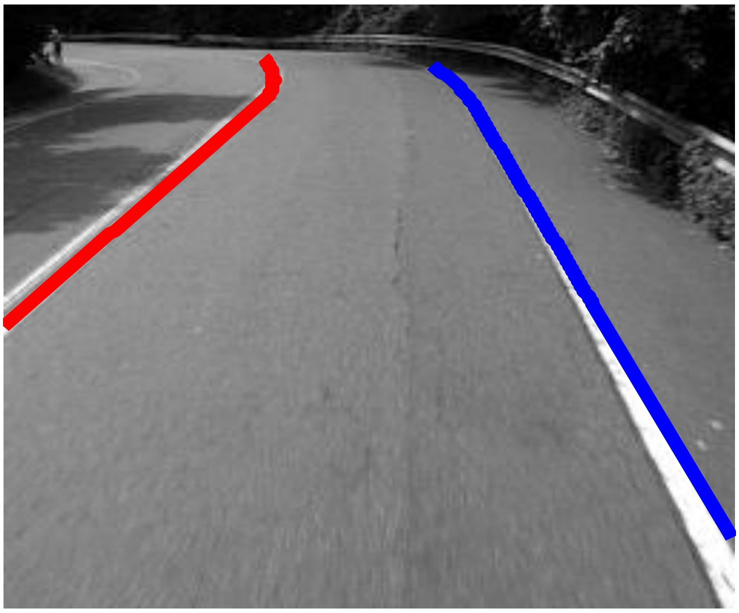


Figure 2.6: Detected lanes.

Chapter3

LANE TRACKING APPROACHES

Lane Tracking is primarily used to enhance the computation efficiency of the lane detection algorithm by maintaining the previous information of how the states have evolved over time so as to have an estimate of the future states. Usually this involves a prediction step and a measurement step. In case of Lane Tracking, prediction step involves moving the detected lines by a certain amount in the image, based on ego-vehicle velocity or by making some assumptions. In measurement step, a new measurement is obtained which will then be used to correct the predicted lane marking positions. Significant research has already been done in Lane Tracking. The most frequently used lane tracking with linear parabolic model. Sections below give a brief overview of literature in this direction.

3.1 Linear Parabolic Model

For the initial tracking, a linear model is chosen, since it provides a robust automatic trackings. However, this model is not apparently suitable for the curved roads. Simpler models demand less computational power and are usually less sensitive to noise. On the other hand, models with more degrees of freedom can provide a more accurate fit to the lane boundary, but are more likely to be affected by image artifacts. In this thesis a lane boundary model that is flexible enough to follow curved roads, robust with respect to road variations (noise, shadows, and weak lane markings) is utilized. Additionally it that can provides information about lane orientation and curvature.

3.1.1 The Proposed Lane Boundary Model

The following model $f(x)$ for lane boundary has been considered:

$$F(\chi) = \begin{cases} a + bx & if\ x > x_m \\ c + dx + ex^2 & if\ x \le x_m \end{cases} \quad (3.1)$$

Where x_m represents the border between near and far fields, as shown in Fig 3.1. These conditions imply that:

$$\begin{cases} a + bx_m = c + dx_m + ex_m{}^2 \\ b = d + 2ex_m \end{cases} \quad (3.2)$$

We can solve this system for the variables c and e, obtaining:

$$c = a + \frac{x_m}{2}(b - d) \quad and \quad e = \frac{1}{2x_m}(b - d) \quad (3.3)$$

Replacing these values back into equation (3.4), we obtain:

$$f(x) = \begin{cases} a + bx & if\ x > x_m \\ \frac{2a + x_m(b-d)}{2} + dx + \frac{(b-d)}{2x_m}x^2, & if\ x \le x_m \end{cases} \quad (3.4)$$

Hence, we need only three coefficients $(a, b\ and\ d)$ to describe our lane boundary model. To determine these parameters, we apply a minimum weighted square error approach, fitting the proposed model to the detected edges.

3.1.2 Fitting the Lane Model

A lane boundary was detected in the previous frame, and the corresponding LBROI was obtained [25]. The edge image $|\nabla I(x, y)|$ of the current frame is computed within the LBROI. Although most of the edges will be related to the lane boundary, some edges related to noise, road texture or other structures would also appear. To remove these undesired edges, we apply an adaptive threshold based on the mean magnitude M_{mean} of the edges. More specifically, we remove all the edges with magnitudes smaller than 0.5 M_{mean}. Let g(x,y) denote the thresholded edge image:

$$g(x,y) = \begin{cases} |\nabla I(x,y)|, & if\ |\nabla I(x,y)| \geq 0.5\, M_{mean} \\ 0, & otherwise \end{cases} \tag{3.5}$$

It should be noticed that this adaptive threshold is not affected by varying illumination Conditions, and does not require any a priori information about the contrast between the road and the image background.

Let $x((x_{ni}; y_{ni})$, for $i = 1,\ldots,$ m, denote the m coordinates of the non-zero pixels of the thresholded edge image $g(x,y)$, belonging to the near field, and $M_{ni} = g(x_{ni}, y_{ni})$ the respective magnitudes.

Analogously, let $(X_{fj};Y_{fj})$ and $M_{fj} = g(X_{fj}, Y_{fj})$ for $j = 1,\ldots,n$ represent the same characteristics for the n edge pixels in the far field. Fitting the lane model (10) to the edge data results in a linear system with three unknowns and $n+m$ equations:

$$\begin{cases} a + bx_{ni} = y_{ni} & i = 1,\ldots.m \\ \dfrac{2a + x_m(b-d)}{2} + dxf_j + \dfrac{(b-d)}{2x_m} x^2 f_j = yf_j & j = 1,\ldots,n \end{cases} \tag{3.6}$$

Typically, $(n+m)$ will be much greater than three, and this system will not admit an exact solution. However, we can find an approximated solution such that a specific error measure is minimized. Assuming that edges related to lane boundaries usually have larger magnitudes than edges related to other irrelevant structures (such as noise, road texture, etc.), we propose a quadratic error weighted by the respective edge magnitudes:

$$E = \sum_{i=1}^{m} M_{n_i} \left[y_{ni} - f(x_{ni})\right]^2 + \sum_{j=1}^{n} M_{f_j} \left[yf_j - f(xf_j)\right]^2 \tag{3.7}$$

This error is minimized when the following 3 ×3 linear system is solved:

$$A^T W A c = A^T W b,$$

Where

$$
\mathbf{A} = \begin{bmatrix} 1 & x_{n_1} & 0 \\ \vdots & \vdots & \vdots \\ 1 & x_{n_m} & 0 \\ 1 & \frac{1}{2x_m}\left(x_{f_1}^2 + x_m^2\right) & -\frac{1}{2x_m}\left(x_{f_1} - x_m\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & \frac{1}{2x_m}\left(x_{f_n}^2 + x_m^2\right) & -\frac{1}{2x_m}\left(x_{f_n} - x_m\right)^2 \end{bmatrix},
$$

$$
\mathbf{W} = \begin{bmatrix} M_{n_1} & & & & & \\ & \ddots & & & & \\ & & M_{n_m} & & & \\ & & & M_{f_1} & & \\ & & & & \ddots & \\ & & & & & M_{f_n} \end{bmatrix},
$$

$C = [a,b,d]^T$ *and* $b = [y_{n1}, \ldots, y_{nm}, y_{f1}, \ldots, y_{fn}]^T$ [7].

It should be noticed that A^TWA is a symmetric matrix. Hence, only a triangular portion (upper or lower) of the matrix must be computed, reducing the computation burden. Figure (3.1) shows the proposed model fit to the second frame of our video sequence, where the dashed line indicates the border between the far field and the near field. Since this Figure illustrates a straight portion of the road, the parabolic part of the model is approximately linear.

Figure 3.1: The dashed line indicates the border between the near and far fields [7].

3.2 Connected Component Method

Besides tracking the lane by using linear parabolic model, connected component method will employ as a novel method to improve the tracking process. The proposed method could help to removes some noises in terms of shadow, discreet lines etc. In this method detected lines which are tracked by using linear parabolic model will be improved by connected component. Once the main lines are detected, some parts would outcome which owns an intersection with the main lines which are beneficial in finding the sequence lines and pixels. This method considers the region that has more than half of the pixels in common, also find the same pixels with the main line and defining them as the main line. Chiefly this phenomenon will assist us in detection and tracking process of the curved parts due to the fact that at the end of the road the lines will approach to a single point and will cause some errors in the process. This method has the ability of recognizing the main lines and removing the pixels that do not have any intersection or less than half of the available pixels in common. Meanwhile, the studied system owns some imperative advantages including detection of the discreet lines and satisfactory performance on the presence of normal shadow on the road. In some cases, if the lines are not evident or there is not enough bright, system can detect the lines precisely. Moreover, its detection ability of the curved lines is in high level.

On the other hand, there are some cases that cannot operate as well as the latter mentioned functions. These kinds of drawbacks may occur on severe shadows, fully curved lines at the end of roads, and in the case of rail boundaries. The comparison of the performance of the considered system for the three mentioned weaknesses has been investigated and compared later in section 4.7.

Chapter 4

SIMULATION RESULTS

4.1 Introduction

In this chapter we provide simulation results for our proposed hybrid lane-detection and tracking algorithm. Simulations were based on frames from custom recorded videos and also from videos which were previously taken and used by Carnegie Mellon University (CMU) researchers. During the experiments we have used two custom AVI videos which we refer to as ROAD-22 and ROAD-23. ROAD-22 has been taken at sunrise, ROAD-23 was shot during the midday. We have paid attention to take the videos at different times during the day so that our hybrid detection and tracking algorithm had to cope with different lighting conditions. From CMU we chose to use the video frames provided in directories “run2a”, “run2b” and “may30_90”. Frames in “run2a” directory has lots of cast shadows on the surface of the road where we need to detect the lanes and is a more challenging video. Individual frames belonging to above mentioned sequences can be downloaded at http://vasc.ri.cmu.edu//idb/html/road. All the detection and tracking programs needed were developed using the MATLAB platform.

For the custom videos ROAD-22 and ROAD-23, the percentages of correctly detecting the lanes were 98% and 75%. For CMU directories run2a, run2b and may30-90, the percentages were 93%, 90% and 95%. In the following sections we will describe our processing steps and also provide our results.

4.2 Lane Detection Based Hough Transform

This section describes how the Hough Transform described earlier in section 2.3 of this thesis is used to extract the lane boundaries in the near field of the video frames given the grayscale images (obtained by a color transformation from RGB to gray). We also provide an example by selecting a single video frame from the Carnegie Mellon University directory run2a (frame#-05) and processing the given frame as required. Subsections below each give details of the various steps we need for lane detection and tracking.

4.2.1 RGB to Grayscale Conversion

The first processing step involves converting the RGB color video frame to gray-scale. Conversion to gray-scale can be done using "lightness method" which takes an average of the least and most prominent colors, or via the "simple average" method, (R+G+B)/3, or through the use of the "luminosity" method. Luminance is luminous intensity per unit projected area and is related to power of the signal. It is usually denoted as Y. Strictly speaking, luminance should be expressed in units such as candelas per meter squared. In practice luminance is often normalized to 1 or 100 units with respect to the luminance of a specified or implied white reference. For example, a studio broadcast monitor has a white reference whose luminance is about 100 cd /m^2, and Y= 1 refers to this value. So in image science, luminance is more properly called relative luminance. Luminance can be computed as a *linear* combination of red, green, and blue primary components (tri-stimulus components). In this thesis RGB to grayscale conversions were done using the relative luminance method and the gray scale image was computed as in (4.1)

$$I_{gr} = 0.21R + 0.72G + 0.07B \quad (4.1)$$

The weights have been chosen such that green light contributes the most to the intensity perceived by humans and blue the least. The weights in (4.1) are also used by most of the contemporary video cameras.

Reducing the image to grayscale not only offers less computational complexity (keep in mind we have 30 frames per second of video), it also reduces the sensitivity to scene condition (under different illuminations). Figure 4.1 depicts the color conversion from RGB to Gray Scale using Frame #-05 from “run2a” directory.

(a) (b)

Figure 4.1: Converting RGB frames to grayscale using luminance method (a) Original frame, (b) Gray scale of original frame.

4.3 Three-Stage Lane Boundary Detection

After obtaining a grayscale of the original frame, a three-stage lane boundary detection is performed. These three stages are namely: 1) vertical mean distribution, 2) lane region analysis, and 3) lane marking detection.

4.3.1 Vertical Mean Distribution

At the preliminary stage, a traffic scene image *I(x,y)* is divided into sky region and road region by means of vertical mean distribution. Vertical mean distribution is measured by averaging the grayscale values of each row on *I(x,y).*A threshold value is acquired through a minimum search along the vertical mean curve. Generally the place at which this first minimum occurs marks the location of the horizon line. Due to the fact that sky region usually possesses higher intensity values than the road pixels there will be generally a big jump in the mean values obtained which allows us to select a proper threshold. Using the selected threshold for a given frame the position of the horizon line is determined and then a line is superimposed on the particular frame as shown in Figure 4.2.

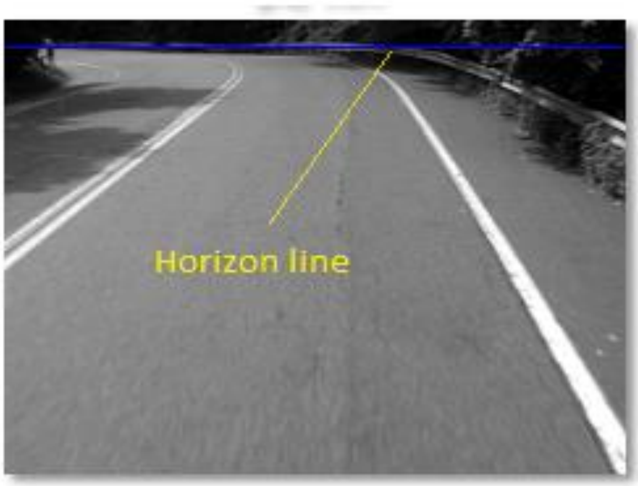


Figure 4.2: Determining the position of the horizon line.

4.3.2 Lane Region Analysis

Lane region analysis is performed to further classify road region and lane markings. Usually, the bottom region in a road image contains road pixels and car pixels (which we have in our custom video road22). By avoiding a few rows from the bottom of the image we can make sure that car parts are not misclassified as road regions. Afterwards, one needs to apply the lane region analysis steps which are listed below:

(i) Skip 30-60 rows from bottom to avoid the possible existence of inner part of a vehicle at the edge of the image.

(ii) Obtain a binary image through a local threshold process followed by some morphological operations

(iii) Find an appropriate threshold and apply edge detection using Canny operator

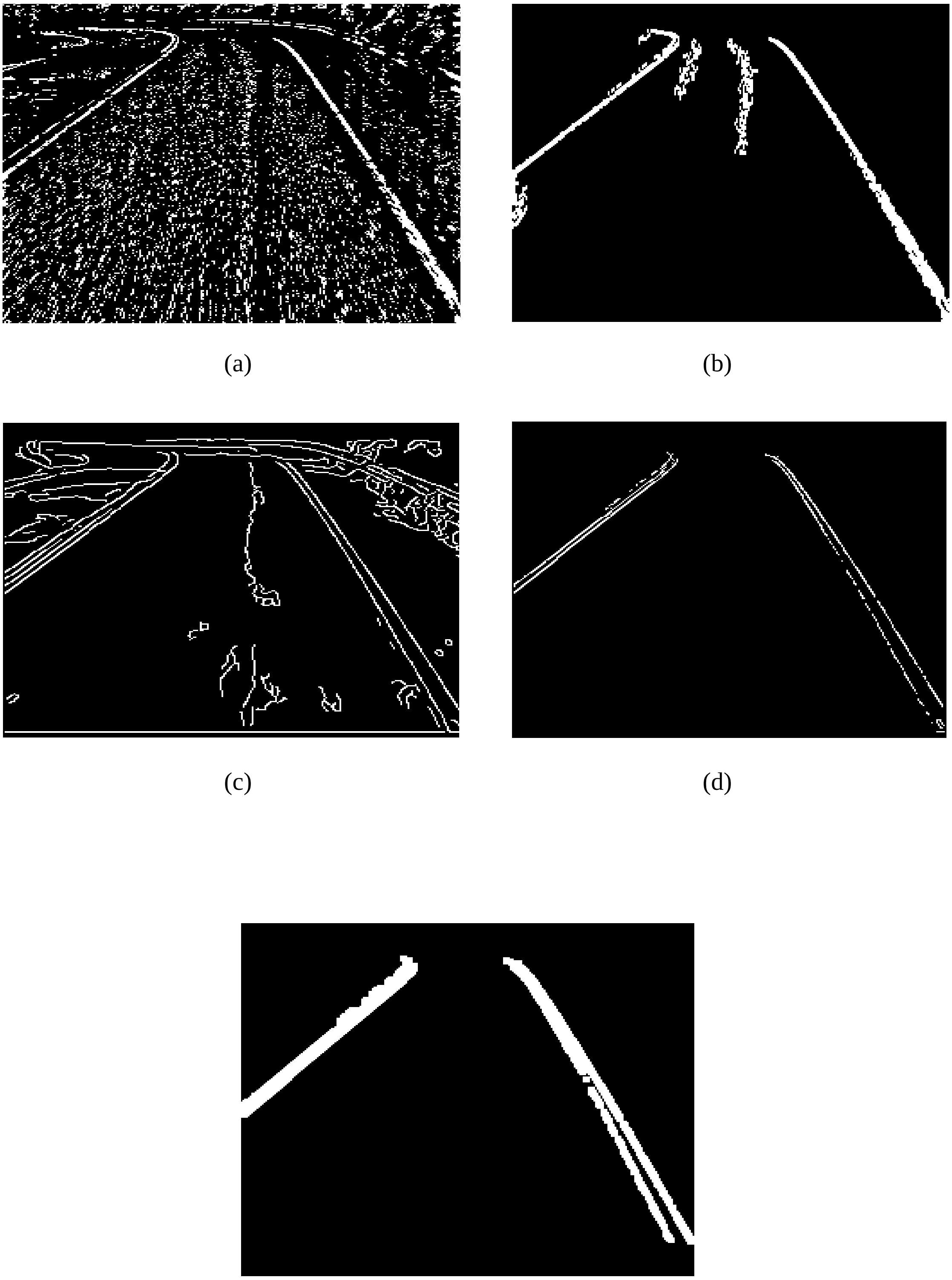
(iv) Create a new edge mask by combining the binary image with the edge image through a logical AND operation

(v) Apply some morphological operations (removal of small components etc.)

(vi) Finally apply Hough transform to the final binary mask to choose the longest possible lane

The threshold required for edge detection is obtained by analysis of a number of rows belonging to the road image in the near field.

Figure 4.3 and the sub-images therein show the binary image before and after morphological operations, the edge image and the mask obtained by ending the two results in subfigures (a),(b) (c) and (d). Figure 4.3 (e) depicts the final edge image after morphological operations before it is passed to the Hough transform block.



(e)
Figure 4.3: (a)-(e) Lane Region Analysis

4.3.3 Analysis of Detected Edge Points for Possible Lane Marks

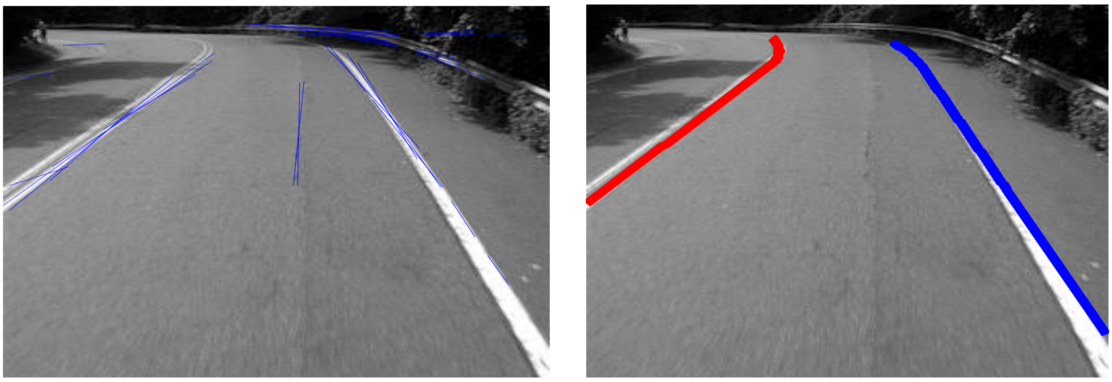
In this step detected edge points will be process by using the Hough Transform to find the candidate lanes in the edge image. The procedure we follow is as follows:

(i) We take the Hough transform and find the peaks of the transform

(ii) Adjacent peaks are treated as a single peak

(iii) Since each peak correspond to a particular line we choose the edge points that belong to the line corresponding to the peak chosen

Applying the above steps to the edge image helps us obtain the lines which are candidates for marking the left and right lanes. Once all lines are detected the ones which have approximate horizontal and vertical orientations are removed. Among the remaining, the lines which they have positive angles denote the left boundary and the lines with negative angles denote the right boundary. For marking the lanes we choose the group of edge pixels which constitute the longest straight lines in the near field. Figure 4.5 (a) depicts the Hough transform detected lines and subfigure (b) shows the marked lanes. For the far-field a parabolic model is applied while marking the lanes.



(a) (b)

Figure 4.4: Detection of possible lane marks and lane marking
(a) All the detected lines with Hough Transform, (b) Marked lanes

4.4 Lane Tracking

There are strong reasons behind the need to apply a proper tracking module to any image processing system. First of all, most of the image processing operations are time consuming which can cause uncertainty, due to miss identification. Secondly, it reduces the computational cost by reducing the search area and hence the corresponding pixel operations. Finally, it heavily cancels out the noise by discarding other parts of the image and therefore the accumulative effects of the noise is reduced. Next to the detection stage, lane tracking system is implemented to restrict the edges searching area on the subsequent frames. Linear parabolic model is mainly used for tracking linear and curved part of the road. A linear model is applied to follow the straight line in the near field since a parabolic model is used to fit the far field. Hence, due to the fact that in far field as curved part of the road in some cases the system have to improve. Connected component has been applied to improve the model especially for the curved part in far field.

4.4.1 Linear Parabolic Model

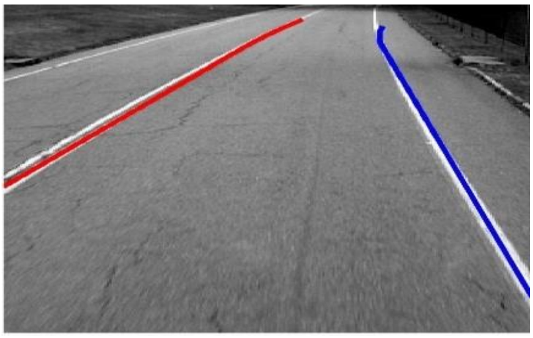
As mentioned earlier in section 3.1, for the initial detection, a linear parabolic model is chosen. The reason is the fact that, it provides a robust automatic detection. Simpler models demand less computational power and are usually less sensitive to noise. In the current thesis, a lane boundary model which is approximately flexible for following the roads, is applied. Moreover, it is robust with respect to several road conditions in terms of noise, shadow, and weak lane markings. Also it provides information about lane orientation and curvature.

4.4.2 Proposed Connected Component Function

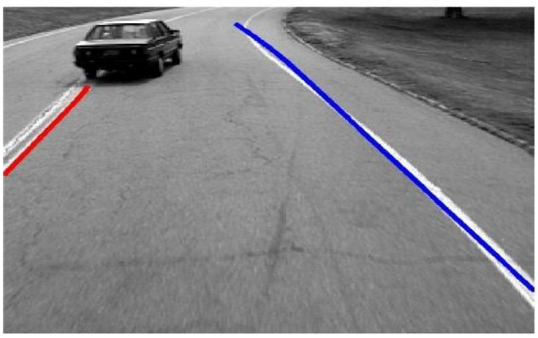
After applying linear parabolic model, connected component method were added to it for improving the performance of the model. As stated in section 3.2, proposed method is capable of finding the pixels owning intersection with the detected line. This means that, it keeps the desired pixels which are components of the previously detected line and add the pixels that they have more than half intersection by the line. After that it removes the pixels that they does not have any intersection by the considered line. In the next sections we will illustrate the experimental results of our methods which they consists of many frames and choose them from each samples that we have.

4.5 Lane Detection for Video Frames from CMU Database

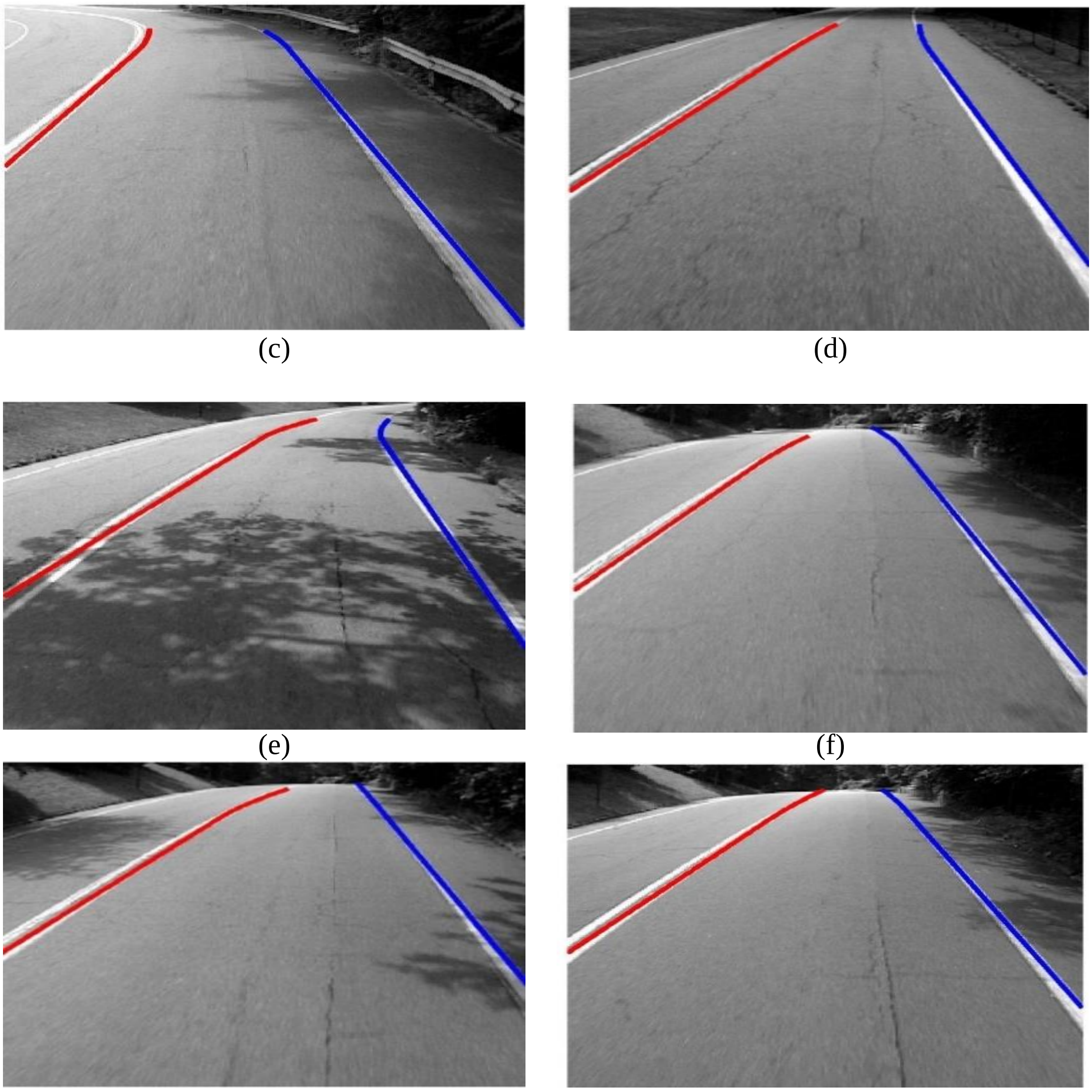
The current section covers the experimental results for our hybrid lane detection method with three sets of different video frames obtained from Carnegie Mellon University (CMU). These three sets include the frames provided in directories "run2a", "run2b" and "may30_90" in the CMU database. Frames in "run2a" directory has cast shadows from trees and cracks on the surface of the roads which make detection of correct lane locations more challenging. Figure 4.5 shows some sample frames marked with detected left and right lanes for the run2a set. When all frames in the ruun2a set are processed we see that our hybrid detection and tracking algorithm has 96.4 % accuracy in detecting the left lane and 98.2 % accuracy for the right lane. In Figure 4.6 we show the incorrectly or partially detected frames of set run2a.



(a)



(b)



(c) (d)

(e) (f)

(g) (h)

Figure 4.5: Lane detection and tracking for **run2a** set.

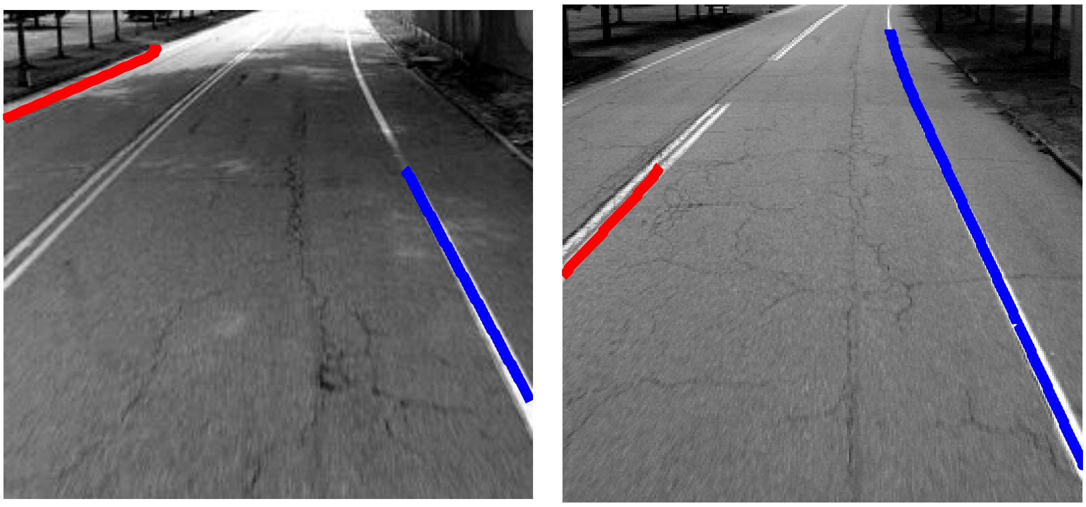
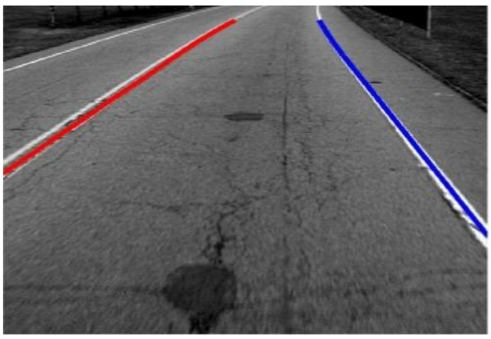


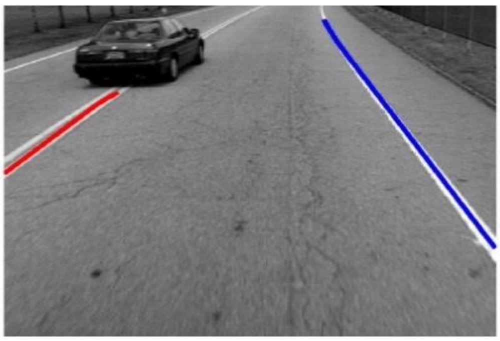
Figure 4.6: Incorrect detections on set **run2a**

The result of lane detection using data set run2b has been provided in Figure 4.7. For the run2b set which contains a total of 57 frames (not all are shown in figure 4.8) our hybrid algorithm has 91.2% accuracy in detecting the left lane (52 correct detections) and 92.9% accuracy for the right one (53 correct detections). Some instances where our lane detection algorithm has failed to correctly detect and mark the left or right lanes have been provided in Figure 4.8.

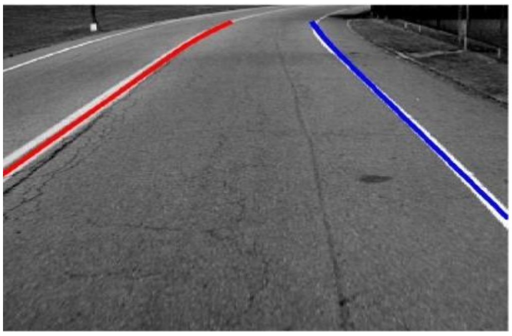
Similarly, the lane detection results using the May30-90 set has been provided in Figure 4.9. The May30-90 set from CMU database contains a total of 37 frames and our hybrid lane detection technique is able to mark all the left and right lanes in each frame successfully.



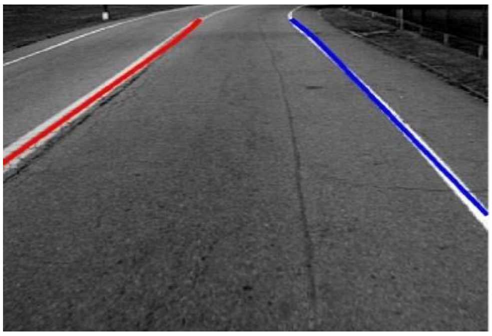
(a)



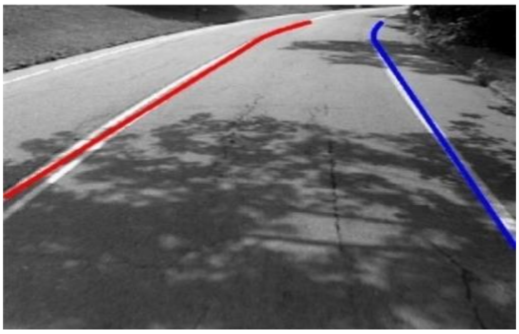
(b)



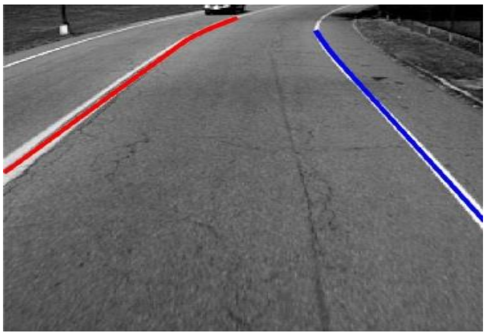
(c)



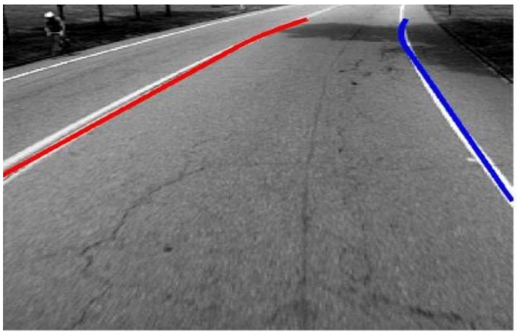
(d)



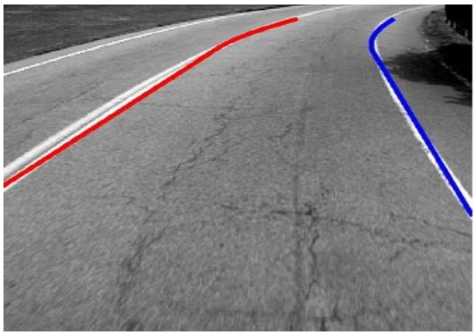
(e)



(f)



(g)



(h)

Figure 4.7: Lane detection and tracking for **run2b** set.

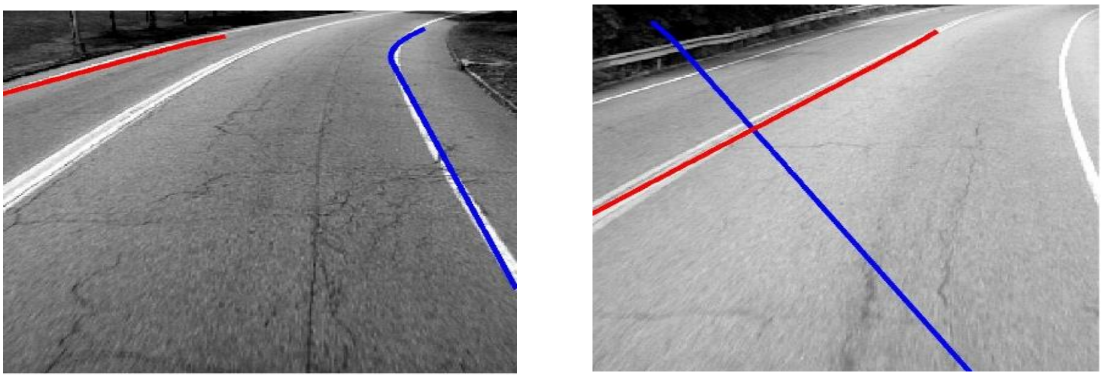


Figure 4.8 Incorrect detection on set **run-2b**.

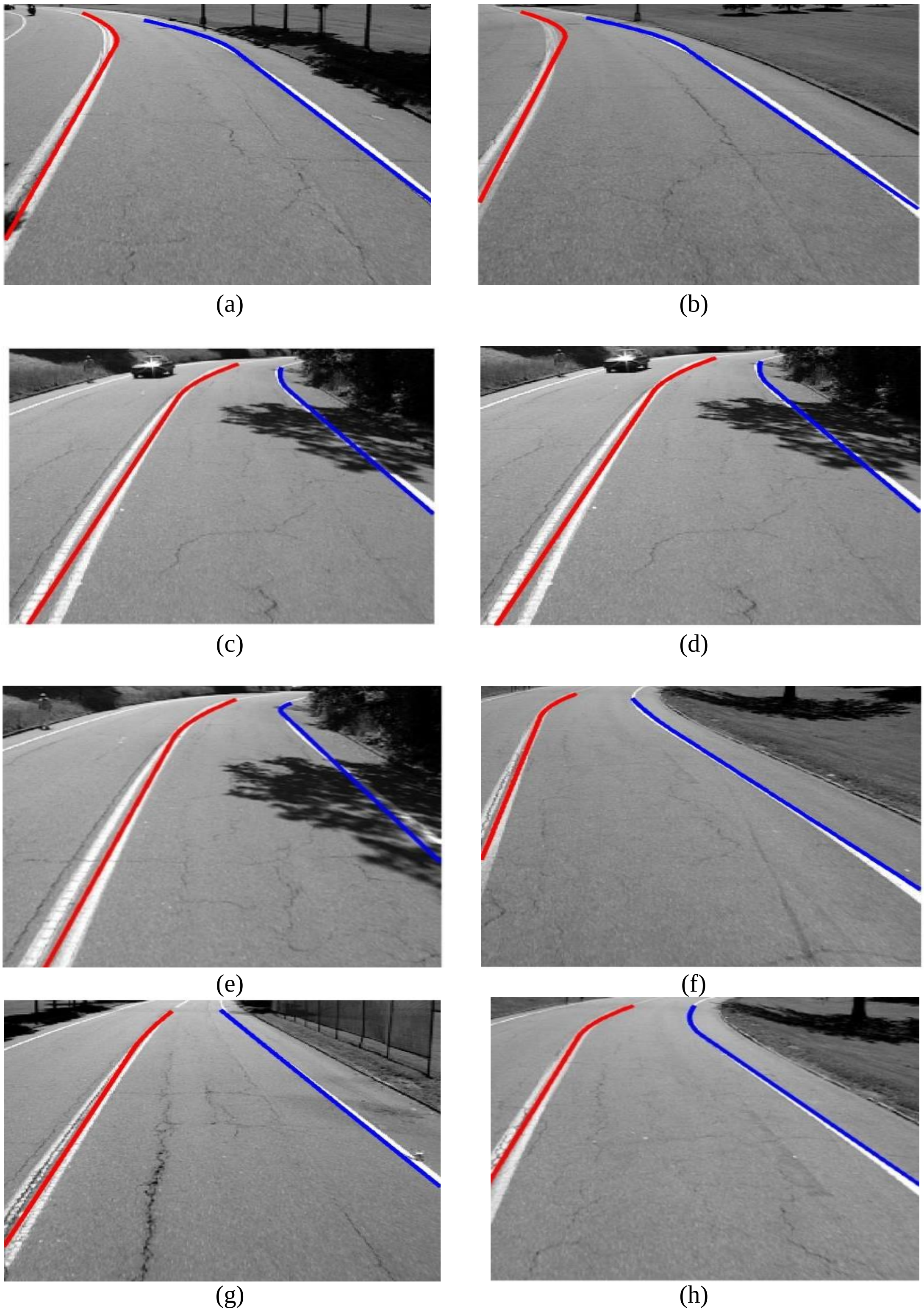
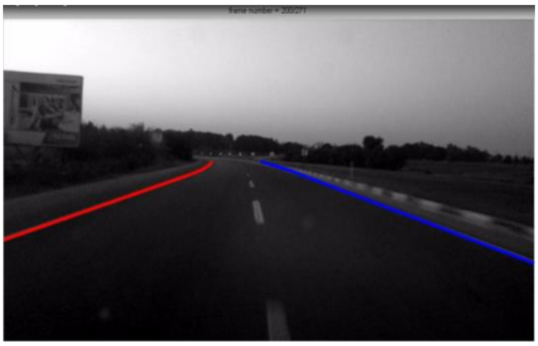


Figure 4.9: Lane detection and tracking for set **may30-90**

4.6 Lane Detection Using Custom-Recorded Video Sequences

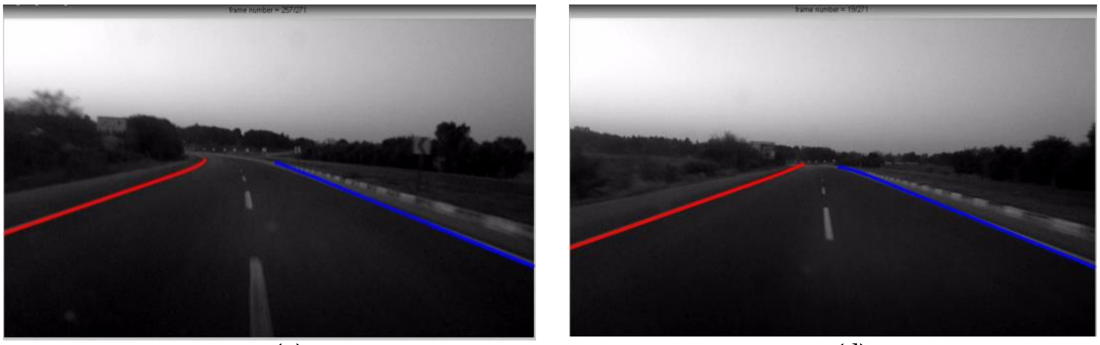
During the experiments we have also used two custom AVI videos which we refer to as ROAD-22 and ROAD-23. ROAD-22 has been taken at sunrise, ROAD-23 was shot during the midday. We have paid attention to take the videos at different times during the day so that our hybrid detection and tracking algorithm had to cope with different lighting conditions. Figures 4.10 and 4.12 respectively depict some sample frames with lanes marked on each side of the road. The sequence ROAD-22 has a total of 272 frames and ROAD-23 has 205 frames. Our experiments point out that while using the ROAD-22 sequence the accuracy of our hybrid lane detection method was 94.8 % for the right lane and 98.52 % for the left lane. Similarly for ROAD-23 sequence the accuracy of detecting the right and left lanes correctly were 95.6% and 99.5% respectively.

Figures 4.11 and 4.13 depict some frames where our algorithm fails to correctly detect and mark the lanes in sequences ROAD-22 and ROAD-23.

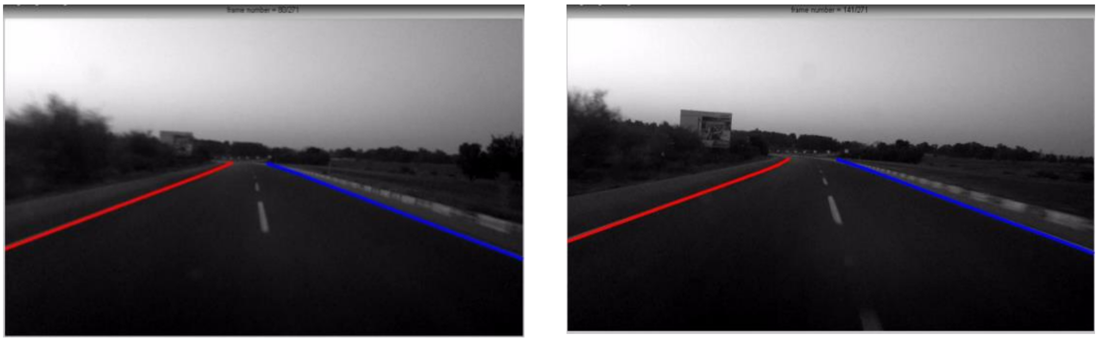


(a)

(b)



(c) (d)



(e) (f)

Figure 4.10: (a)-(f) Lane Detection using ROAD-22 custom video.

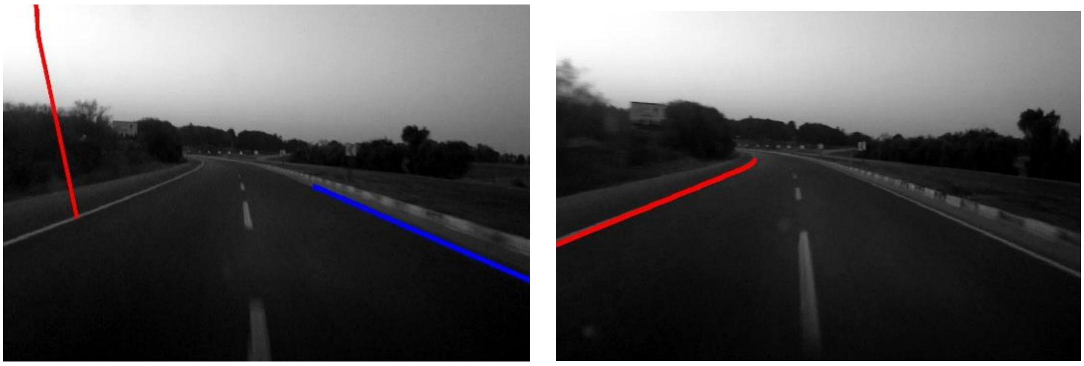
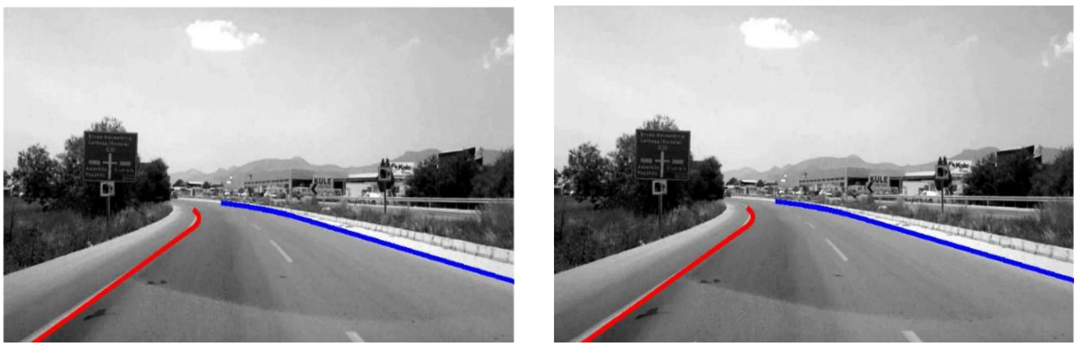
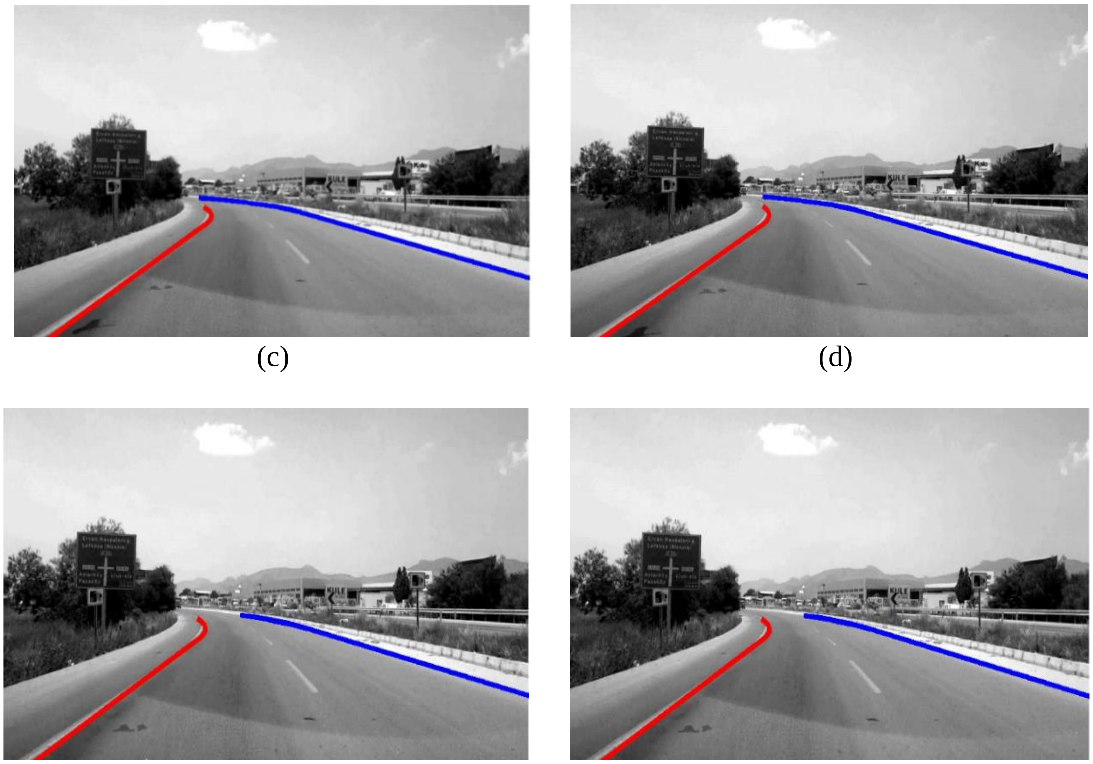


Figure 4.11: Incorrect detections on ROAD-22 custom video.



(a) (b)



(c) (d)

(e) (f)

Figure 4.12: (a)-(f) Lane detection using ROAD-23 custom video.

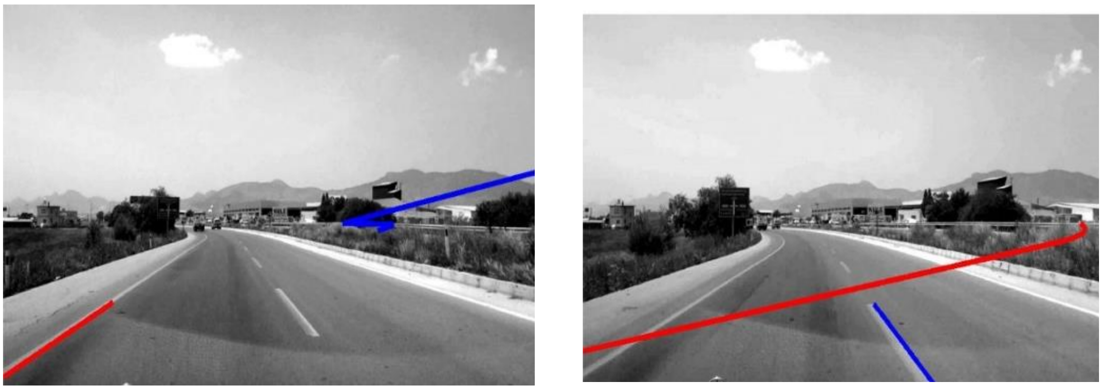


Figure 4.13: Incorrect detection on ROAD-23.

4.7 Feasibility of Real Time Operation With Respect to Speed

In order to assess the computational complexity of the proposed hybrid lane detection and tracking algorithm we first computed the time required to fully processing a single frame of size (256×240). This was achieved by using the

MATLAB stopwatch functions 'tic' and 'toc'. For a (256×240) frame, processing time was found to be around 11 seconds. With the knowledge that implementing the algorithm using a high level language would reduce the time requirement by around tenfold we can assume here that the time requirement would be around 1.1 seconds. The video camera that was used in our experiments had a frame rate of 30 frames per second. This would mean that time between consecutive frames is 33.33ms. To operate in real-time one would need to complete all required processing in a time less than this value. Since 1.1s is larger than 33.33ms at first it may appear as if real-time processing is not possible. Fortunately since in the real world the car only moves a short distance in 33.33ms we do not need to process the entire frame that the camera provides due to inverse perspective projection (refer to Figure 4.14).

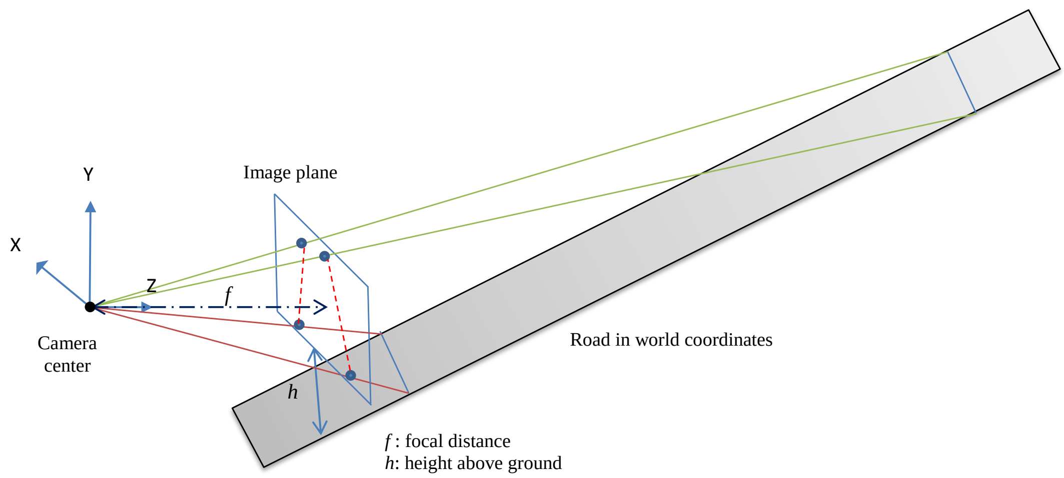


Figure 4.14: Projection from world coordinates to image space

For example, when we assume a vehicle moving at a speed of 36km/h, in 33.33ms the distance it would cover on the ground would be around 0.33 meters. If we play safe and even assume that the front of the vehicle moves twice as far this would mean covering 0.66m in the physical world. If the road the camera is focused at is 30m long then processing time required would be around 1/45th of the time required for the full frame. Since the time required for the full frame was 1.1s this would mean a processing time of 24.4ms. Because this value is below the 33.33ms frame separation time one could say that real-time processing is possible at 36km/hr. When the above computations are repeated for a speed of 72km/h the processing time would become 48.8ms and since this value is more than the frame separation time we can no more do real time processing.

Chapter 5

CONCLUSION

Lane detection and tracking is an important application of Intelligent Transport System. To avoid victims and number of accidents in heavy traffic countries like USA, China, Malaysia, UK, IRAN, where it becomes difficult for the driver to exact location and detection of line and cars especially during cloudy environment than it is important to make Intelligent Transport System more robust and as well in other way lane detection and tracking is one of important future application of auto drive vehicle.

Till now so many different vehicle companies and researchers have used different ways and develop different algorithms under different conditions to make the Intelligent Transport System more robust to noise and detection but they usually operate under certain type of scene conditions and more complex to implement under different conditions. As in case of lane detection we described and implemented the Hough Transform and edge detection by using canny algorithm. Then we described linear parabolic model to make Hough Transform more efficient not only for finding lines but also it provides tracking the lines. Furthermore, connected component method add to the system due to the fact that tracking the far field. In far field lines become unclear and it is hard to track them. Our proposed method provide the system to track the lines much better in this case.

Our proposed algorithm was implemented in MATLAB R2012a on a DELL (Vostro 1500) computer with CPU of Intel ® Core™ 2 Duo with the processor frequency of 2.0 GHz and RAM of 4.00 GB. We have processed captured custom videos (ROAD-22 and ROAD-23) and also Carnegie Mellon university video frames (**run2a**, **run2b** and **may30-90**) for this thesis. As future work, a formal evaluation of the performance should be made. Moreover, the robustness of the algorithm will be tested by applying it to other video sequences. In the car analysis, we believe that the detection algorithm is robust enough, but results could be improved by using more advanced tracking methodologies for the future work.

REFERENCES

[1] Assidiq, A. A. M., Khalifa, O. O., Islam, M. R. and Khan, S., "Real time lane detection for autonomous vehicles," in *Proc. of the Int. Conf. on Computer and Communication Engineering , ICCCE08*, 2008, pp. 82-88.

[2] Zhou, S. Jiang .Y, Xi . J, Gong . J, Xiong . G, and Chen . H., "A novel lane detection based on geometrical model and Gabor filter," in *IEEE Intelligent Vehicles Symposium,* 2010, pp. 59-64.

[3] Schneiderman, H and Nashman . M., "Visual processing for autonomous driving," *IEEE Workshop on Applications of Computer Vision*, 1992, pp. 164-171.

[4] Litkouhi, B . B, Lee . A . Y, and Craig . D . B., "Estimator and controller design for LaneTrak, a vision-based automatic vehicle steering system," *32nd IEEE Conf on Decision and Control*, 1993, vol.2., pp. 1868-1873

[5] Taylor, C . J, Malik . J, and Weber . J., "A real-time approach to stereopsis and lane-finding," *IEEE Conf on Intelligent Vehicles Symposium*, 1996, pp. 207-212.

[6] Betke, M . Haritaoglu, E, and Davis . L . S., "Highway scene analysis in hard real-time," *IEEE Conf on Intelligent Transportation System, ITSC'97,Nov* 1997, pp. 812-817.

[7] Jung, C . R, and Kelber . C . R., "A robust linear-parabolic model for lane following," *17th Brazilian Symposium on Computer Graphics and Image Processing*, 2004, pp. 72-79.

[8] Fletcher, L . Petersson, L, and Zelinsky . A., "Road scene monotony detection in a fatigue management driver assistance system," *Proc. of IEEE on Intelligent Vehicles,* 2005, pp. 484-489.

[9] Hsieh, J . W, Shih-Hao . Y, Yung-Sheng . C, and Wen-Fong . H., "Automatic traffic surveillance system for vehicle tracking and classification," in *IEEE Transaction on Intelligent Transportation Systems,* vol. 7, 2006, pp. 175-187.

[10] Maire, F and Rakotonirainy . A., "Analysis of Driving Session Videos by Reverse Temporal Order Processing," *Int. Conf. on Computer Graphics Imaging and Visualisation*, 2006, pp. 255-261.

[11] McCall, J . C, and Trivedi . M . M., "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE Transactions on Intelligent Transportation Systems,* vol. 7, no.1, 2006, pp. 20-37.

[12] Isa, K , "Experimental Studies on Dynamics Performance of Lateral and Longitudinal Control for Autonomous Vehicle Using Image Processing," *IEEE 8th Int. Conf. on Computer and Information Technology Workshops,* 2008, pp. 411-416.

[13] Borkar, A, Hayes . H, and Smith . M . T., "An efficient method to generate ground truth for evaluating lane detection systems," *IEEE Int. Conf. on Acoustics Speech and Signal Processing ,ICASS'10,* 2010, pp. 1090-1093.

[14] Wen-Chang, C and Chia-Ching . C., "The development of the automatic lane following navigation system for the intelligent robotic wheelchair," in *IEEE Int. Conf. on Fuzzy Systems, FUZZ'11*, 2011, pp. 1946-1952.

[15] Benligiray, B, Topal . C, and Akinlar . C., "Video-Based Lane Detection Using a Fast Vanishing Point Estimation Method," *IEEE Int. Symposium on Multimedia,* 2012, pp. 348-351.

[16] Gopalan, R,Tsai . T, Shneier . M, and Chellappa . R., "A Learning Approach Towards Detection and Tracking of Lane Markings," *IEEE Transactions on Intelligent Transportation Systems,* vol. 13, 2012, pp. 1088-1098.

[17] Bertozzi, M, and Broggi . A., "GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Transactions on Image Processing*, vol. 7, Issue .1, 1998, pp. 62-81.

[18] Mei, C . Jochem . T, and Pomerleau . D., "AURORA: a vision-based roadway departure warning system," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* 1995, vol.1, pp. 243-248.

[19] Qing, L, Youngjoon . H, and Hernsoo . H., "Real-Time Lane Departure Detection Based on Extended Edge-Linking Algorithm," 2nd *Int. Conf. on Computer Research and Development* , 2010, pp. 725-730.

[20] Guo, K, Li . N, and Zhang . M., "Lane Detection Based on the Random Sample Consensus," *Int. Conf on Information Technology, Computer Engineering and Management Sciences (ICM)*, 2010, pp. 38-41.

[21] Yim, Y . U, and Se-young . O., "Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems,* vol. 4, No. 4, Dec 2003, pp. 219-225.

[22] Assidiq, A . A . M, Khalifa . O . O, Islam . R, and Khan . S., "Real time lane detection for autonomous vehicles," *Int. Conf. on Computer and Communication Engineering, ICCCE'08*, 2008, pp. 82-88.

[23] Dagao, D, Meng . X, Qian . M, Zhongming . H, and Yueliang . W., "An improved Hough transform for line detection," *Int. Conf. on Computer Application and System Modeling, ICCASM'10*, 2010, Vol. 2, pp. -354-357.

[24] Jung, C . R, and Kelber . C . R., "A robust linear-parabolic model for lane following," *17th Brazilian Symposium on Computer Graphic and Image Processing*, 2004, pp. 72-79.