Estimation of Vehicle's Lateral Position via the Lucas-Kanade Optical Flow Method

JIANN-SHIOU YANG
Department of Electrical Engineering
University of Minnesota
Duluth, MN 55812
USA
jyang@d.umn.edu http://www.d.umn.edu/~jyang

Abstract: - The use of rumble strips on roads has proven to be an effective means of providing drivers lane departure warning (LDW). However, rumble strips require an infrastructure and do not exist on a majority of roadways. Therefore, it is desired to develop an effective virtual rumble-strip LDW system. However, before developing such a system, it is essential that we know the vehicle's lateral characteristics; in particular, the vehicle's lateral position and speed. In this paper, we focus on using image processing via an in-vehicle camera to estimate the vehicle's lateral position and speed. The lateral position is determined by finding the vehicle's heading angle via a homography and the Lucas-Kanade optical flow techniques; while the lateral speed is determined via the heading angle and the vehicle's On Board Diagnostic (OBD) System forward speed data. This paper describes our approach and findings in some detail together with suggested improvements. The advantage of our approach is that only the minimal set of information to characterize the vehicle lateral characteristics is needed and thus, making it more feasible in a vehicle application.

Key-Words: - Heading angle, homography, histogram equalization, Harris corner, feature selection and tracking, optical flow.

1 Introduction

Roadway departure fatalities including run-off-the-road (ROR) and head-on fatalities are a series problem in the United States. According to the National Motor Vehicle Crash Causation Survey (NMVCCS) data [1], ROR crashes contribute to a large portion of fatalities and serious injuries to motor vehicle occupants and over 95% of the critical reasons for single-vehicle ROR crashes were driver-related. The category of the critical reasons attributed to 27.7% driver performance errors (e.g., poor directional control, overcompensation), followed by 25.4% driver decision errors (e.g., too fast for curve/conditions), 22.5% critical non-performance errors such as sleeping, physical impairment and 19.8% driver recognition errors (e.g., internal and external distractions). Moreover, statistics data indicated that 70% of ROR fatalities occur on rural highways and about 90% occur on two-lane roads [2]. The most common approach to prevent single vehicle lane departure is the use of rumble strips [3, 4] on road shoulders. Actually, the use of rumble strips on roads has proven to be an effective means of providing drivers departure warning [5]. But rumble strips require an infrastructure and are not available on all roadways. Furthermore, rumble strips presents a difficult issue of where to establish the rumble-strip distance threshold (see the headline news: "Rumble Strips Unpopular in Other Minnesota Counties, Too", "Roadsides Rumbles Cause a Grumble", *Duluth News Tribune*, September 12, 2010; "County Vows to Reduce Rumble", *Duluth News Tribune*, September 29, 2010). Development of various techniques such as lane departure warning (LDW) systems can improve traffic safety significantly. An LDW system should be able to detect when the driver is in danger of departing the road and then trigger an alarm to warn the driver early enough to take corrective action. Various LDW techniques including the "time-to-lane-crossing" (TLC) method can be found in the literature (e.g., [6-12]). Other commercially available forward looking, camera-based systems in vehicles that use algorithms to interpret video images to estimate vehicle state and roadway alignment include AutoVue (Iteris's device), the Infiniti Lane Departure Prevention (LDP) system, the Volvo's and BMW's LDW systems, the Lexus LS 460 multi-mode Lane Keeping Assistant (LKA) system, etc. AutoVue is already being offered as a factory-installed option on Freightliner's Century and Argosy trucks.

Automakers offering a lane departure warning system on some of their models include Audi, Infiniti, BMW, Cadillac, Volvo and Buick.

Before developing an effective LDW system, it is essential that we know the vehicle's lateral characteristics, in particular, the vehicle's lateral position and speed. Therefore, this paper mainly focuses on determination of the vehicle's heading angle, which in turn is used to determine the vehicle's lateral position and speed and thus, the vehicle's lateral characteristics. Several existing methods to determine either the vehicle's heading angle or vehicle's lateral offset can be found in the literature. Suppachai *et. al.* used a compass sensor as a feedback sensor in their double-loop controller design to measure the heading angle of the vehicle [13]. Y. Jintao used a magnetoresistive sensor to determine the heading angle for a mobile robot [14]. Pinto and Reid proposed finding the heading angle and offset as a pose recognition problem, where each combination of heading angle and offset was considered as a pose, and they applied the principal component analysis to find the parameters [15]. In [16], Jung and Kelber used a linear-parabolic model to detect lane boundaries and the linear part of their model was then used to obtain an estimation of the vehicle's lateral offset. Tamiya *et. al.* explored the spread spectrum method via infrared light to measure vehicle's lateral distance [17]. Other methods such as the optical flow based method (e.g., [18, 19]), the focus of expansion (FOE) method (e.g., [20]), etc. can also be found in the literature.

A recent method to estimate the lateral position of the vehicle based on matching of top-view images from a sequence of moving camera images inside the vehicle has been reported by Teshima *et. al.* [21]. It is based on the so-called plane projective transform (i.e., the homography) between the ground and the image plane of the present and next frame. Homographies are obtained by the computation of the rotation matrix according to the information of the calibrated camera, the vehicles' speed, and the angle of vehicle's direction. The movement obtained from each frame is accumulated to obtain the lateral position of the camera and thus, the vehicle's lateral position from its initial position. The authors in [21] claimed that the effectiveness of their approach has been demonstrated by researchers from Mitsubishi Fuso Truck & Bus Corporation and Keio University in Japan. In addition, their approach does not rely on extraction of features such as lines, rail, or lane markers, but based on matching of warped top-view images between two consecutive frames. In this paper, we use an image-based optical flow technique to determine the lateral characteristics of the vehicle without knowledge of lane markers and GPS information. Although the basic concept of our approach is similar to that of the Teshima's method [21], our approach is quite different. Instead of using a hypothesis of guessing the best candidate having the minimum "sum of absolute difference" (SAD) value among several possible candidate heading angles (with calculated SAD) at each frame, we use the feature selection and tracking of the consecutive top-view images via the pyramidal Lucas-Kanade optical flow method [18, 19] to determine the heading angle. Other potential application techniques/methods to predict the vehicle's heading angle based on its previous data and/or partial knowledge such as the Kalman filtering [22], the agent based modeling under partial knowledge settings [23], the autoregressive integrated moving averaging approach [24], etc. can be found in the literature. In this paper, the OpenCV software package [19], a strong focus on real-time applications originally develop by Intel, is used to process the image conversion and transformation. We use a homography to remove the perspective effect from the front-view images captured by an in-vehicle camera to convert them to top-view (or the bird's eye view) images. The resulting image represents a top view of the road region in front of the vehicle, as it was observed from a significant height. We then use feature selection and tracking from these top-view images, via the Lucas-Kanade optical flow technique [18] to determine the vehicle's heading angle. In addition, we use the Shi and Tomasi's method to determine good corners (features) [25]. The heading angle of the vehicle and thus, the lateral displacement can be calculated by evaluating the relationship between those tracked features. This lateral position estimation is being used as one of the input variables to develop a decision making operation algorithm for lane departure warning in our Phase II study [26]. An alternative technique that we can consider is a rule-based fuzzy logic approach (e.g., [27, 28]) which can be potentially used to establish the LDW operations.

This paper is organized as follows. Section 2 describes the vehicle's lateral characteristics and the heading angle determination. The homograpgy and histogram equalization of image conversion and processing are discussed. The schematic diagram and implementation flow chart to determine the vehicle's heading angle are then followed. In

Section 3, the Lucas-Kanade optical flow method including the Region of Interest (ROI), the Harris corners detection and tracking is introduced in some detail. Section 4 presents our road test results and findings together with some discussion. Finally, Section 5 gives the conclusion.

2 Vehicle's Lateral Characteristics

In order to develop an effective LDW system, we need to know the vehicle's lateral characteristics. That is, the lateral position and its tendency in lateral movement/momentum. To identify the lateral position of the vehicle, it is essential that we determine the vehicle's heading angle and its lateral speed. The vehicle's lateral speed can be determined once we know its heading angle and forward speed. Let *y*-axis be the vehicle's longitudinal axis, *x*-axis be the perpendicular lateral axis, u be the vehicle's forward speed and θ its heading angle (i.e., the angle between the vehicle moving direction and its longitudinal y-axis). Then, given the vehicle's current position (x, y) it is easy to see that the vehicle's lateral speed can be expressed as $\dot{x} = u \sin(\theta)$, where $\dot{x}$ means the time derivative of *x*. For small θ, we have $\dot{x} \approx u\,\theta$ and u can be further replaced by $u \approx \bar{u}$, where $\bar{u}$ is the vehicle's average forward speed. Therefore, the vehicle's lateral speed is determined by $\dot{x} = \bar{u}\,\theta$. Note that the assumption of small heading angle and constant forward speed are justifiable for highway driving conditions and the vehicle's forward speed can be measured via the vehicle's On-Board Diagnostics (OBD)-II.

2.1 Vehicle Forward Speed Access

On-Board Diagnostic systems can be found in most cars and light truck on the road today. During the 1970' and early 1980' manufactures started using electronic means to control engine functions and diagnose engine problems. OBD-I was placed on vehicles in the early 1980s. Through the years on-board diagnostic systems have become more sophisticated. Nowadays, its second generation diagnostic system called OBD-II provides almost complete engine control and also monitors parts of the chassis, body and accessory devices, as well as the diagnostic control network of the vehicle. According to the information from [29], all light duty vehicles (e.g. less than 8,500 pounds) sold in North America since 1996, as well as medium duty vehicles (e.g. 8,500-14,000 pounds) beginning in 2005, and heavy duty vehicles (e.g. greater than 14,000 pounds) beginning in 2010, are required to support OBD-II diagnostics using a standardized data link connector. For details regarding the OBD-II standardization and data interface, please refer to [30].

As mentioned above almost all the vehicles manufactured in the United States today are required, by law, to provide an interface for the connection of diagnostic test equipment. The data transfer on these interfaces follow several standards, but none of them are directly usable by PCs or smart devices. Since it is important to send the forward speed data to the computer, we need to access the vehicle's OBD-II. The schematic diagram shown in Fig. 1 is used to access the vehicle's forward speed in real time and thus, we can determine its lateral speed $\dot{x}$ via $\dot{x} \approx \bar{u}\,\theta$ (where the heading angle θ will be explained later). Note that in Fig. 1, the ELM 327 chip, a multi-function OBD-II interpreter, is used to act as a bridge between OBD ports and a standard RS232 interface for systems that use the ISO 15765-4 CAN, SAE J1850 PWM, SAE J1850 VPW, ISO 9142-2, ISO 14230-4 and SAE J1939 protocols.

2.2 The Homography

To determine the vehicle's heading angle, we used the image-based technique via an in-vehicle camera. Several existing methods such as the optical flow based method [18, 19], the lane marker based method [31], the focus of expansion (FOE) method [20] can be found in the literature. In our study, in order to determine the vehicle's heading angle more accurately, we converted the front-view images captured by the camera to their corresponding top-view images (i.e., the bird's eye view images) via a homography. Note that the conversion to top-view image involves removing the perspective effect in the front-view image. That is, re-sample the incoming image and re-mapping each pixel in the captured image toward a different position and producing a new two-dimensional array of pixels. The resulting image represents a top view (or the bird's eye view) of the road region in front of the vehicle, as it was observed from a significant height. In the field of computer vision, any two images of the same planar surface in space are related by a homography. There are many practical applications (e.g., image registration, rectification, computation of camera motion (rotation, translation)) between two images. In this study, we mainly deal with the accumulated lateral displacement of an in-vehicle camera in order to find the vehicle's lateral position, Once camera rotation and translation have been extracted from an estimated homography matrix, this information will be later used in our feature

detection and tracking of the consecutive top-view images via the Lucas-Kanade optical flow method [18] to determine the vehicle's heading angle.

Our image processing and conversion are conducted via an in-vehicle camera. It is known that camera lens can cause distortion and distortion causes "fisheye" effect; therefore, the camera needs to be calibrated to remove such a distortion. This step is important and it is also necessary to produce a "good" quality of front-view images and thus, provides a better feature selection/detection and tracking in image processing. However, we found that if a non wide-angle camera is used then the nonlinear distortion near the boundaries of the images can be minimized. Thus, the camera calibration together with a needed distortion file becomes unnecessary. Therefore, in determining the vehicle's heading angle, we use a low cost Microsoft H5D LifeCam Cinema CMOS image sensor to capture front-view images. Figure 2 shows a snapshot taken from this camera and Figure 3 shows its corresponding top-view image after the homography transformation. We used the Open Source Computer Vision Library (OpenCV) to conduct the homography [19]. This OpenCV software package is a library of programming functions mainly aimed at real time computer vision applications which we will use later in our image processing including the homography and optical flow method.

2.3 Histogram Equalization

The contract of the images in a scene captured by the camera is very important to improve the quality and success rate of feature detection and tracking when using the optical flow techniques. Therefore, the images need to be preprocessed to enhance their contrast. Histogram equalization can increase the global contrast of images, especially when the usable data of the image is represented by close contrast values. Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast. The mathematics behind histogram equalization involves mapping one distribution (the given histogram of intensity values) to another distribution (a wider and ideally uniform distribution of intensity values). Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. This method is useful in images with backgrounds and foregrounds that are both bright or both dark. The brightness distribution is equalized thereby increasing the contrast of the image. Therefore, we applied histogram equalization to the front-view images so that there is sufficient texture variations to detect good features later. The effect of applying the contrast enhancement technique to Fig. 2 can be seen in Fig. 4, where Fig. 4 (a) shows the original image (i.e., Fig. 2), while Fig. 4 (b) shows the image after contrast enhancement.

3. Image Feature Selection and Tracking

Optical flow is an approximation of the image motion based on local derivatives in a given sequence of images. It is the distribution of apparent velocities of movement of brightness patterns in an image. In a two-dimensional (2D) plane it specifies how much each image pixel moves between adjacent images. Sequences of ordered images allow the estimation of motion as either instantaneous image velocities or discrete image displacements. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement. There are several methods available for determining optical flow including the phase correlation method [32], the block-based method [33], the differential methods [18, 34] and discrete optimization method [35], etc. The differential methods are based on partial derivatives of the image signal (e.g., the Lucas-Kanade method [18], the Horn-Schunck method [34]). In this study, we focus on the Lucas-Kanade optical flow method to determine the vehicle's heading angle via the top-view image processing. In the following, we give a brief concept introduction about this approach.

3.1 The Lucas-Kanade Optical Flow Method

The Lucas-Kanade (LK) algorithm is based on the following three assumptions: (a) brightness constancy, (b) temporal persistence (implies "small movements"), and (c) spatial coherence [18]. The first assumption means that a pixel from the image of an object in the scene does not change in appearance as it moves from frame to frame. That is, the brightness of a pixel does not change as it is tracked from frame to frame. The second assumption simply means that the image motion changes slowly in time, i.e., the motion is slow relative to the frame rate. The third assumption implies that neighboring points in a scene stay neighbors. In other words, the neighboring points belong to the same surface, have similar motion, and project to nearby points on the image plane.

The Lucas-Kanade optical flow method tries to calculate the motion between two image frames which are taken at times t and t + Δt at every pixel position [19]. This method is also called differential since it is based on the local Taylor series approximations of the image signal; that is, it uses partial derivatives with respect to the spatial and temporal coordinates. Let's consider a 2D case (3D or higher-order dimensional cases are similar). Let the brightness intensity $I(x, y, t)$ is the center pixel in a n × n neighborhood and moves by Δx, Δy in time Δt to $I(x+ \Delta x, y+ \Delta y, t+ \Delta t)$. The brightness constancy assumption is the requirement that pixels in one tracked patch look the same over time. Therefore, we have the following expression between the two image frames

$$I(x,y,t) = I(x+\Delta x, y+\Delta y, t+\Delta t) \quad (1)$$

Assuming the movement to be small, the image constraint at $I(x, y, t)$ with the Taylor series expansion becomes

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + \dots \quad (2)$$

From the assumption (a), we have

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad (3)$$

and, thus

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \quad (4)$$

which results in

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0 \quad (5)$$

where V_x, V_y are the x and y components of the velocity or optical flow of $I(x, y, t)$ and $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$, rewritten as I_x, I_y and I_t respectively, are the derivatives of the image at (x, y, t) in the corresponding directions. That is,

$$I_x V_x + I_y V_y = -I_t \quad (6)$$

or

$$\nabla I^T \cdot \vec{V} = -I_t \quad (7)$$

where the spatial intensity gradient $\nabla I = [I_x \ I_y]^T$ (T means transpose) and $\vec{V} = [V_x \ V_y]$ (the image velocity or optical flow at pixel (x, y) at time t). This is the 2D motion constraint equation with two unknowns for any given pixel. Clearly, this equation can not be used to obtain a unique solution for the 2D motion at that point. However, if a local patch of pixels moves coherently, then we can easily solve for the motion of the central pixel by using the surrounding pixels to set up a system of equations. This spatial coherence is exactly the 3rd assumption proposed by the Lucas-Kanade optical flow method. That is, neighboring points in a scene belong to the same surface, have similar motion, and project to nearby points on the image plane (i.e., neighboring points stay neighbors). The Lucas-Kanade optical flow implemented a weighted least-squares (LS) fit of local motion constraint equation for $\vec{v}$ in each small spatial neighborhood Ω by minimizing

$$\sum_{x,y\in\Omega} W^2(x,y)[\nabla I(x,y,t)\bar{v} + I_t(x,y,t)]^2 \quad (8)$$

where $W(x, y)$ is a weighting function (typically chosen as 2D Gaussian) that gives more influence to constraints at the center of the neighborhood than those at the periphery. The solution to the above minimization reduces to

$$\bar{V} = [A^T W^2 A]^{-1} A^T W^2 b \quad (9)$$

where, for N pixels (i.e., for a n × n neighborhood N = n^2), $(x_i, y_i) \in \Omega$ at time t and

$$A = [\nabla I(x_1,y_1), \cdots, \nabla I(x_N,y_N)]$$

$$W = diag[W(x_1,y_1), \cdots, W(x_N,y_N)]$$

$$b = -[I_t(x_1,y_1), \cdots, I_t(x_N,y_N)]^T \quad (10)$$

where *diag* means the diagonal matrix. The solution to the motion constraint equation can be solved in closed form when $A^T W^2 A$ is nonsingular (i.e, when it is of rank 2) and

$$A^T W^2 A = \begin{pmatrix} \sum W^2(x,y) I_x^2(x,y) & \sum W^2(x,y) I_x(x,y) I_y(x,y) \\ \sum W^2(x,y) I_y(x,y) I_x(x,y) & \sum W^2(x,y) I_y^2(x,y) \end{pmatrix} \quad (11)$$

where all sums are taken over pixels (x, y) in the neighborhood Ω.

In our study, homographies are obtained by the computation of the rotation matrix according to the information of the calibrated camera, the vehicle's speed, and the angle of vehicle's direction. The movement obtained from each frame is accumulated to obtain the lateral position of the camera, and thus, the vehicle's lateral position from its initial position. Instead of using a hypothesis of guessing the best candidate among several possible candidate heading angles with calculated SAD at each frame [21], we used the feature selection and tracking of the consecutive top-view images via the Lucas-Kanade optical flow method [18, 19] to determine the heading angle.

3.2 Region of Interest (ROI)

The feature selection/detection and tracking are based on the top-view images obtained. At each time frame, a set of "good" features (or "corners") are determined. Trackable (and reliable) features are points that contain enough information to be picked from the current frame to the next frame. They should have brightness constancy, sufficient texture variations, and do not deform much over time [19]. These "corners" can be found by calculating the eigenvalues of the image's 2nd moment matrix, and larger eigenvalues (above a preset threshold) imply good feature. To reduce real-time computation, a Region of Interest (ROI) instead of the whole region in a top-view image is chosen. For instance, for the top-view image shown in Fig. 3, we choose its ROI to be a rectangular box image, a 300 pixel × 165 pixel sample of the image, centered to be straight in front of the vehicle showing the width of the lane about 12 feet. This ROI image, as shown in Fig. 5, starts right after the vehicle's hood on the bottom of Fig. 3 and goes up to show about 10 feet in front of the vehicle. Notice the right and left lane markers in Fig. 5. We want to find good features in the limited area instead of using the whole images. The ROI region is just wide enough to fit the width of a normal lane on a road or highway. We only use the partial image shown in ROI to greatly speed up the computations, thus allowing the algorithm to run in real time more effeciently. In addition, we don't need to worry about the possible distortion effects created by converting to the top-view images since these only arise around the edges of the original image.

3.3 Harris Corners Detection and Tracking

A point of our interest in an image is a point which has a well-defined position and can be robustly detected and tracked. This means that an interest point can be a corner but it can also be an isolated point with local intensity maximum, or minimum. Many trackable points are called "corners", and the words "corner" and "feature" are sometimes used interchangeable in literature. if we choose a point that is unique (i.e., a "good" feature) in a image frame then we have a pretty good chance of finding that point again in the next image frame. In practice, the point or feature we select should be unique, or nearly unique, and should be parameterizable in such a way that it can be compared to other points in another image.

The most commonly used definition of a corner was provided by Harris [36]. This definition relies on the matrix of the second-order derivatives $(\partial^2 x, \partial^2 y, \partial x \partial y)$ of the image intensities. That is, we take the second-order derivatives of images, taken at all points in the image, to form a new "second-derivative images" (or a new Hessian image). This terminology comes from the Hessian matrix around a point, which is defined as

$$\begin{pmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial y^2} \end{pmatrix} \tag{12}$$

The autocorrelation matrix of the second derivative images over a small window around each point is defined as

$$M(x,y) = \begin{pmatrix} \sum_{-k \le i,j \le k} w_{i,j} I_x^2(x+i, y+j) & \sum_{-k \le i,j \le k} w_{i,j} I_x(x+i, y+j) I_y(x+i, y+j) \\ \sum_{-k \le i,j \le k} w_{i,j} I_x(x+i, j+j) I_y(x+i, y+j) & \sum_{-k \le i,j \le k} w_{i,j} I_y^2(x+i, y+j) \end{pmatrix} \tag{13}$$

where $w_{i,j}$ is a weighting function (e.g., Gaussian weighting). The Harris corners are points in the image where the autocorrelation matrix of the second derivatives has two large eigenvalues. The advantage of considering only the eigenvalues of the autocorrelation matrix is that they are invariant to rotation, which is important because objects that we are tracking might rotate as well as move. However, instead of using the Harris's method, we use the Shi and Tomasi's method to determine good corners (features), which result as long as the smaller of the two eigenvalues is greater than a minimum threshold [25].

To explain the corner detection and tracking, let's consider the two consecutive top-view images in

Fig. 6 and their corresponding ROIs shown in Fig. 7. In Fig. 7, the black dots at time i are the Harris corners to be tracked at frame i+1 and the dots identified at frame i+1 are to be tracked at frame i+2. We used a pyramidal approach of the Lucas-Kanade optical flow technique [18, 19] together with the OpenCV to detect the large motions which gave the feature displacement as shown in Fig. 8. In this figure, the moving vectors show how the features selected moved from frame i and i+1. The OpenCV software package [19], a strong focus on real-time applications, is used to process the image conversion and transformation. The schematic diagram showing this image-based sensing approach is given in Fig. 9.

3.4 Heading Angle

Base on the feature displacement, the coordinates of the features in two consecutive frames can then be determined (i.e., by superimposing two ROIs and comparing where these selected points moved). Note that only those points, forming a sunset of points selected from ROI, with increasing y-coordinate are tracked. Averaging the total angles associated with these selected features (with respect to the *x*-axis) gives us the vehicle's heading angle at that time instant.

Consider two consecutive top-view images at frame i and i+1. Let p be the number of points selected in the ROI at frame i, $x_{i+1,j}$ be the *x*-coordinate of the j^{th} point at frame i+1, $x_{i,,j}$ be the *x*-coordinate of the j^{th} point at frame i, $y_{i+1,j}$ the *y*-coordinate of the j^{th} point at frame i+1, and $y_{t,,i}$ be the *y*-coordinate of the j^{th} point at frame i. Also define $(\Delta x)_j = x_{i+1,\,j} - x_{i,j}$, $(\Delta y)_j = y_{i+1,\,j} - y_{i,j}$ and let n represent the number of points with $(\Delta y)_j < 0$. Then, If $(\Delta x)_j > 0$ let $\theta_j = -(90^o - \beta_j)$ (left turn); If $(\Delta x)_j < 0$ let $\theta_j = (90^o - \beta_j)$ (right turn); and if $(\Delta x)_j = 0$ then set $\theta_j = 0$. Note that β_j is defined

$$\beta_j = \frac{180^o \times \alpha_j}{\pi} \tag{14}$$

where

$$\alpha_j = \tan^{-1} \left| \frac{\Delta y_j}{\Delta x_j} \right| \tag{15}$$

and $j = 1$ to n. The heading angle is determine by a simple averaging as follows

$$\theta = \frac{1}{n} \sum_{j=1}^{n} \theta_j \tag{16}$$

The entire process and the concept were tested on the roads. Except for some outliers, overall speaking they showed good results.

3.5 Determination of the Vehicle's Lateral Position

Based on the real time contrast enhanced top view images, we use the Lucas-Kanade optical flow technique [18, 19] to determine the vehicle's heading angle. The schematic diagram of the signal processing to identify the vehicle's lateral position is shown in Fig. 9. Based on the vehicle's current speed and its heading angle, the algorithm calculates the lateral position relative to its initial position. Note that in Fig. 9 the vehicle's relative velocity can be determined by the vehicle's forward speed multiplied by $\sin(\theta)$, and for small heading angle θ, $\sin(\theta)$ can be approximated by θ; therefore, the lateral speed can be determined by the forward speed multiplied by its heading angle.

The flow chart to implement Fig. 9 is shown in Fig. 10. Note that in Fig. 10 the vehicle "initial" lateral position will be reset as often as possible once the lane markers become available. A mechanism to reset the vehicle's "initial" lateral position via the camera detection of the road edges is being implemented into the operation algorithm in our Phase II study which focuses on developing an LDW system [26].

4 Results and Findings

The vehicle's lateral position is determined by implementing the schematic diagram shown in Fig. 9 with the iterative signal flow operation given in Fig. 10. Note that OpenCV is an open source computer vision library written in C and C++ and runs under Linux, Windows and Mac OS X. There is active development on interfaces for Matlab, and other languages. This software originally was developed for computational efficiency with a strong focus on real time applications. Since its library contains over 2,500 optimized algorithms that span many areas (e.g., medical imaging, user interface, camera calibration, stereo vision, robotics, etc.), we use the open source library functions OpenCV 2.3.1 to implement our operation algorithms. In this section, we discuss about the results and finding during our road tests.

4.1 Road Tests

We conducted road tests to check the accuracy of our operation algorithm in finding the vehicle' heading and thus, its lateral displacement. The tests were conducted in several areas of Duluth, Minnesota with speeds at 50 miles per hour (mph) and 65 mph. A segment of Highway 61 between Duluth and Two Harbors was chosen as a test site with driving speed 65 mph while the tests at 50 mph speed were performed on segments of Martin Road, Lavaque Road, and Jean-Duluth Road.

Figure 11 shows a sample screenshot of one of the many spreadsheets the algorithm generated (with the heading angle data shown in real time) while we conducted the road test on one segment of Jean Duluth Road lying north of Martin Road. The test was conducted at 9:30 am on a sunny morning. During part of this road test a total of 200 image frames were processed over a time period of 20 seconds (i.e., the time between consecutive frames is 100 milliseconds). Note that in Fig. 11, some details about the captured image frames, the points (Harris corners) tracked, the angle changes between those tracked points, and the calculated angles from individual frame features can be seen. The frame comparison code (Z) in Fig. 11 represents that the images Z and Z-1 are compared to see how the orientation of the vehicle has changed from one frame to another. Note that Fig. 11 only shows the initial image frame at Z = 1 and the first 18 of total 29 features tracked at the frame labeled Z = 2. Corner Ax and Ay denote the coordinate points of individual features (Harris corners) found in the image Z-1 that are to be searched for in the next image Z. The algorithm tracks the features (Ax, Ay) in the subsequent frame to find their corresponding new locations (Bx, By). Not all the features found in the image Z-1 could be tracked in the following image. Sometimes the features are not found or it may be that the features did not pass a preset threshold of accuracy. Once the coordinates of the same feature has been found, the angle (Alpha) between them is calculated which tells us how the feature has moved in those two consecutive images. The angle is then determined for each feature and averaged for each Z as given in Eqs. (14-16). For instance, in Z = 2 when the captured images 1 and 2 are compared, 292 points were found in image 1 (i.e., Z = 1) that were considered as good features to track. However, only 29 points could be tracked in the image 2 (only 18 are shown in Fig. 11). Thus angle of these 29 points are estimated and averaged. While averaging, a condition is placed such that if the number of features tracked is below a certain threshold (we have set it to 5) then its averaged angle (Beta) is set to zero. The reason is that sometimes only one or two points may be tracked, and given that the algorithm may not always track features with 100 percent confidence it may skew the result. Thus, for each Z there should be at least 6 features tracked to be averaged, or else the angle is by default assigned zero. After processing a batch of every ten number of Z's, that is at every 1 second, average of Beta angles of that batch is computed which will be our final heading angle at that time with respect to the previous position. In Table 1 it can be seen that after Z = 10 and 20, average of those batches are taken to compute the final heading angle; that is at time t = 1 and 2 seconds the angles are 18.16^o and 8.07^o, respectively. Hence using this heading angle, the lateral displacement of the vehicle deviated from the center of the lane is computed as shown in Table 2 from t = 1 to 18 seconds.

4.2 Results

In this section, we explain the some of the results we found while conducting the tests on Martin Road, Jean Duluth Road, Lavaque Road and Highway 61 outside the city of Duluth limit. As mentioned in Section 4.1, the speed limit on part of Jean Duluth and Lavaque is 50 mph mile, while it is 65 mph on Highway 61. Figure 12 shows the heading angle and lateral position of the vehicle traveling at 50 mph. In Fig. 12, during the first 10 seconds, the heading angle is positive meaning that the vehicle is moving toward its right direction and thus, the vehicle deviates from the center of the lane but less than 2 feet (ft). The following negative angle indicates the vehicle is moving to the left relative to its previous direction. Overall, the deviation from its center is within the range of -0.5 to 1.5 ft in this case.

The results in Table 2 are shown in Fig. 13. As stated above, the test was performed on a small straight section of the road. Ideally, the heading angle should be zero during the course of the test. But, in practice, maintaining a zero angle would be impossible; so a few degrees variations are expected. In Fig. 13, the heading angle and lateral displacement with respect to its previous position is shown at every second. As can been seen, the range of the heading angle lies between 18.7^o and -12.76^o, and the lateral displacement is from 2.97 feet (right) to - 2.11 feet (left).

In order to study the effect of changing the threshold of the feature selection on the heading angle, and thus, the lateral displacement, we analyzed the data with threshold set to 5 and 15. Figures 13 and 14 show the heading angle and lateral position with 50 mph speed but a different threshold (i.e., 5 in Fig. 13 and 15 in Fig. 14). Overall, they look similar. The threshold has some influence on the heading angle but not significant in this case. We see that the vehicle deviates from its center of the lane but remains within ± 4 ft. Figures 15 and 16 show a similar results with a higher 65 mph speed on Highway 61. Again, the deviation from the center of the driving lane lies within the 5 ft range.

To improve the accuracy of the system operation over time, it is important to reset the vehicle's "initial" position as often as possible, as shown in Fig. 10. Therefore, in our Phase II study, a lane detection algorithm was also integrated with the operation algorithm to implement Figs. 9 and 10. We used the Hough transform [37-39] which is a method for findings lines, circles, or other simple forms in an image and it is a feature extraction technique concerning with the identification of lines in the images. We used this technique to detect the lane edges and thus, to reset the "initial" position when the lane markers become available. It can also be used to deal with the road curvature issue.

Note that there are a lot of factors that could cause the discrepancy in the heading angle and thus, the vehicle's lateral displacement. Possible factors that could affect the results are

- Even though those points tracked pass a certain threshold of accuracy, some points might still not have been tracked accurately enough.

- Sometimes the consecutive images may look very similar (i.e, at higher sampling rate and/or higher speed), making it harder for the algorithm to detect changes between those image frames. Results could be that no good distinguishable features be found and tracked or if found, they could be tracked incorrectly as everything looks the same. The histogram equalization could help by increasing the image contrast.

- The default setting of the angles (Beta) to zero could deviate from the actual angle while averaging.

Overall, the results showing the vehicle's lateral position reasonable to reflect the actual driving situation and are also satisfactory. However, refinement of the operation algorithm should be continued via further extensive road tests.

5 Conclusion

The main objective of this study is to determine the vehicle's lateral characteristics in order to lay a foundation for the development of an innovative virtual rumble-strip LDW system. However, before developing such a system, it is essential that we know the vehicle's lateral characteristics; in particular, the vehicle's lateral position and speed. In this study, we used image processing via an in-vehicle camera to estimate the vehicle's lateral position and speed. The lateral position was estimated by determining the vehicle's heading angle, which is obtained via a homography transformation and the Lucas-Kanade optical flow techniques; while the lateral speed was determined via the heading angle and the vehicle's OBD-II port and an OBD-II interpreter. In addition, the histogram equalization was used to enhance the image contrast quality in order to improve the feature selection and tracking. Based on the implemented operation algorithm, we conducted extensive road tests with 50 mph speed on segments of Jean Duluth Road and Lavaque Road, located north of Martin Road, and we also did tests on Highway 61 with 65 mph speed. The detail of our approach was presented together with findings. Possible factors that could cause the discrepancy in the heading angle, and thus the vehicle's lateral displacement, are also given. Overall, the results showing the vehicle's lateral position are satisfactory. Extensive road tests with refinement can be found in our Phase II study [26] where we developed a virtual rumble-strip LDW system using the vehicle's lateral characteristics.. The advantage of our approach given in this paper is that it only needs a minimal set of information to characterize the vehicle lateral characteristics and, therefore, making it more feasible in a vehicle application.

Acknowledgment

The research was funded by the Intelligent Transportation Systems (ITS) Institute, a program of the University of Minnesota's Center for Transportation Studies (CTS). Financial support was provided by the United States Department of

Transportation (USDOT)'s Research and Innovative Technologies Administration (RITA). The project was also supported by the Northland Advanced Transportation Systems Research Laboratories (NATSRL), a cooperative research program of the Minnesota Department of Transportation (MnDOT), and the ITS Institute. The author would like to thank Rini Shrestha and Benjamin Nelson for the experiment set up and road tests.

References:

[1] C. Liu and T. Ye, "Run-Off-Road Crashes: An On-Scene Perspective," (USDOT HS 811 500), National Highway Traffic Safety Administration, 2011.

[2] C. F. Lin and A. G. Ulsoy, "Time to Lane Crossing Calculation and Characterization of Its Associated Uncertainty," *ITS Journal*, Vol. 3, No. 2, 1996, pp, 85-98.

[3] R. E. Turochy, "Shoulder Rumble Strips: Evolution, Current Practice, and Research Needs," *Transportation Research Board Annual Meeting*, Washington, D.C., 2004.

[4] K. Perrillo, *The Effectiveness and Use of Continuous Shoulder Rumble Strips*, National Highway Sleep Foundation, Washington, D.C., 1998.

[5] P. Batavia, "Driver-Adaptive Lane Departure Warning System," Ph.D. Dissertation, Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, 1999.

[6] H. Godthelp, P. Milgram, and G. Blaauw, "The Development of a Time-related Measure to Describe Driving Strategy," *Human Factors*, Vol. 36, No. 3, 1984, pp. 257-268.

[7] J. Hayward, "Near-miss Determination through Use of a Scale of Danger," *Highway Research Record*, No. 384, 1972, pp. 24-34.

[8] D. Pomerleau, "RALPH: Rapidly Adapting Lateral Position Handler," *Symposium of IEEE Intelligent Vehicles*, Detroit, MI, September 1995, pp. 506-511.

[9] D. LeBlanc, G. Johnson, P. Venhovens, G. Gerber, R. DeSonia, R. Ervin, C.-F. Lin, A. G. Ulsoy, and T. Pilutti, "CAPC: A Road Departure Prevention System," *IEEE Control Systems Magazine*, Vol. 16, 1996, pp. 61-71.

[10] C. R. Jung and C. R. Kelber, "A Lane Departure Warning System Using Lateral Offset with Uncalibrated Camera," *Proceedings of the 8th IEEE International Conference on Intelligent Transportation Systems*, 2005, pp. 348-35.

[11] M. Gonzalez-Mendoza, B. Jammers, N. Hernandez-Gress, A. Titli, and D. Esteve, "A Comparison of Road Departure Warning Systems on Real Driving Conditions," *Proceedings of the 2004 IEEE Conference on Intelligent Transportation Systems*, 2004, pp. 349-353.

[12] J. Manigel and W. Leonhard, "Vehicle Control by Computer Vision," *IEEE Transactions on Industrial Electronics*, Vol. 39, No. 2,1992, pp. 181-188.

[13] H. Suppachai, C. Silawatchananai, M. Parnichkun and C. Wuthishuwong, "Double Loop Controller Design for the Vehicle's Heading Control," *Proceedings of the 2009 International Conference on Robotics and Biomimetics*, 2009, pp. 989-994.

[14] Y. Jintao, "The Design of Mobile Robot Heading Control," *Proceedings of the 2010 International Conference on E-Health Networking, Digital Ecosystems and Technologies*, 2010, pp. 9-11.

[15] F. Pinto and J. Reid, "Heading Angle and Offset Determination Using Principal Component Analysis," Paper Number 983113, ASAE Meeting, UILU-ENG-98-7010, 1998.

[16] C. R. Jung and C. R. Kelber, "A Lane Departure Warning System Using Lateral Offset with Uncalibrated Camera," *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, 2005, pp. 348-353.

[17] N. Tamiya, H. Mandai and T. Fukae, "Optical Spread Spectrum Radar for Lateral Detection in Vehicles," *Proceedings of the 4th IEEE International Symposium on Spread Spectrum Techniques and Application*, 1996, pp. 195-198.

[18] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proceedings of Imaging Understanding Workshop*, 1981, pp. 121-130.

[19] G. Bradski and A. Kaebler, *Learning OpenCV Computer Vision with the OpenCV Library*, Cambridge: O'Reilly, 2008.

[20] T. Teshima, H. Saito, S. Ozawa, K. Yamamoto, and T. Ihara, "Estimation of FOE Without Optical Flow for Vehicle Lateral Position Detection," *Proceedings of IAPR Conference on Machine Vision Applications*, 2005, pp. 406-409.

[21] T. Teshima, H. Saito, S. Ozawa, K. Yamamoto, and T. Ihara, "Vehicle Lateral Position Estimation Method Based on Matching of Top-View Images," *Proceedings of the 18th*

International Conference on Pattern Recognition, 2006, pp. 626-629.
[22] J.-S. Yang, "Application of the Kalman Filter to the Arterial Travel Time Prediction - A Special Event Case Study," *Control and Intelligent Systems*, vol. 35, no. 1, pp. 79-85, 2007.
[23] F. Neri, "Agent Based Modeling Under Partial and Full Knowledge Learning Settings to Simulate Financial Markets", *AI Communications*, ISO Press, 2012 (in print).
[24] J.-S. Yang, "Autoregressive Integrated Moving Averaging Modeling for Short-Term Arterial Travel Time Prediction," *WSEAS Transactions on Systems*, vol. 5, no. 4, pp. 751-759, 2006.
[25] J. Shi and C. Tomasi, "Good Feature to Track," *Proceeding of the IEEE Conference on Computer Vision*, 1994, pp. 593-599.
[26] J.-S. Yang, "Development of an Innovative Prototype Lane Departure Warning System Without Knowledge of Lane Markers and GPS Location Information," NATSRL Technical Report, August 2012.
[27] H. Tozan and M. Yagimli, "A Fuzzy Prediction Based Trajectory Estimation", *WSEAS Transaction son Systems*, issue 8, vol. 9, pp. 885-894, 2010.
[28] O. Hachour, "Towards an Approach of Fuzzy Control Motion for Mobile Robots in Unknown Environments," *WSEAS Transactions on Systems*, issue 7, vol. 8, pp. 876-890, 2009.
[29] Wikipedia http://en.wiki pedia.org/wiki/On-board_diagnostics
[30] K. McCord. Automotive Diagnostic Systems, CarTech, 2011.
[31] H. Wang M. Ren, and S. Shao, "Lane Markers Detection based on Consecutive Threshold Segmentation," *Journal of Information and Computer Science*, Vol. 6, No. 3, 2011, pp. 207-212.
[32] E. De Castro and C. Morandi "Registration of Translated and Rotated Images Using Finite Fourier Transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 5, 1987, pp. 700-703.
[33] E. G. Richardson, *H.264 and MPEG-4 Video Compression:Video Coding for Next-generation Multimedia*, John Wiley & Sons, 2003.
[34] B. B. K. P. Horn and B.G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, Vol. 17, 1981, pp. 185-203.
[35] B. Glocker, N. Komodakis, G. Tziritas, N. Navab & N. Paragios, *Dense Image Registration through MRFs and Efficient Linear Programming*, 2008.
[36] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," *Proceedings of the 4th Alvey Vision Conference*, 1988, pp. 147–151.
[37] P. V. C. Hough, "Machine Analysis of Bubble Chamber Pictures," *Proceedings of the International Conference on High Energy Accelerators and Instrumentation*, 1959, pp. 554-556.
[38] R. O. Duda and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Communications of the Association for Computing Machinery*, Vol. 15, 1972, pp. 11-15.
[39] L. Shapiro and G. Stockman, *Computer Vision*, Prentice Hall, 2001.

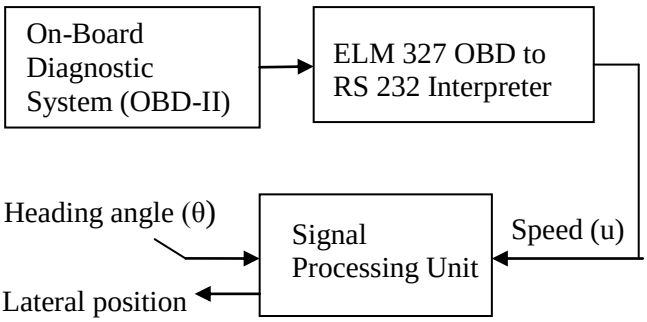

Fig. 1 Measurement of the vehicle's forward speed via OBD-II.

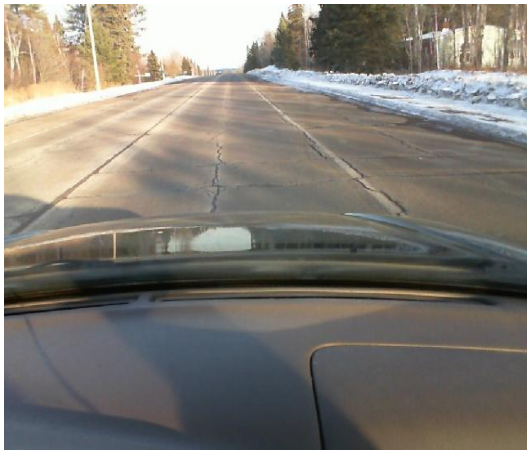

Fig. 2 A Snapshot of a front-view image using the Microsoft H5D camera.

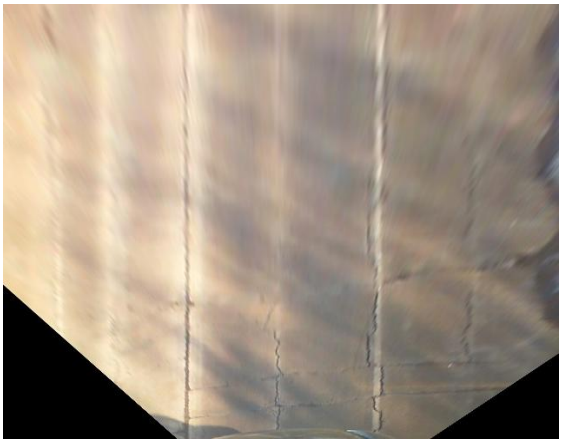

Fig. 3 The front view image after the homography.

(a) (b)

Fig. 4 Effect of histogram equalization (a) before and (b) after.

Fig. 5 The ROI image.

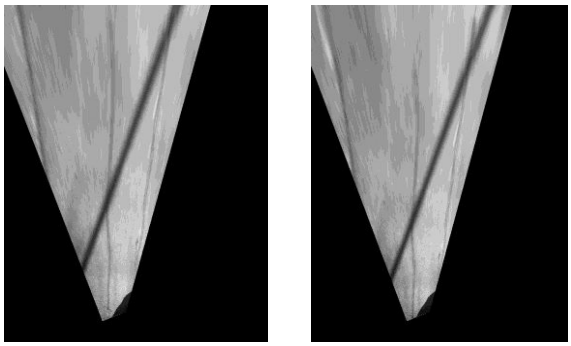

Fig. 6 Top-view images at time i and i+1 after homography.

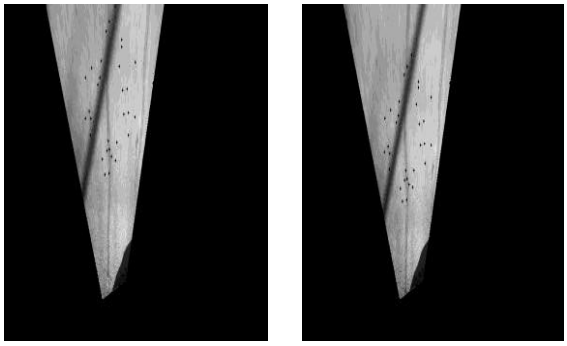

Fig. 7 ROI at time i and i+1.

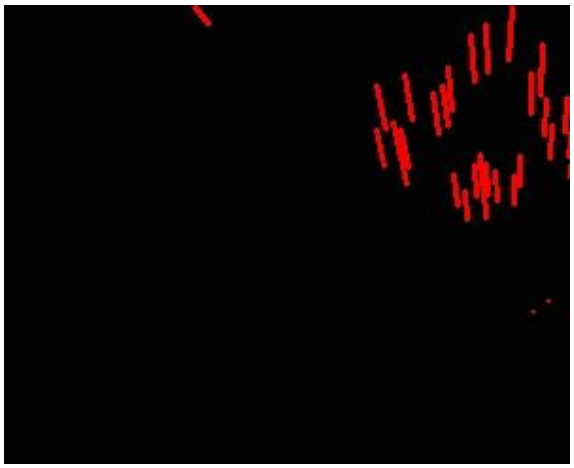

Fig. 8 Feature displacement.

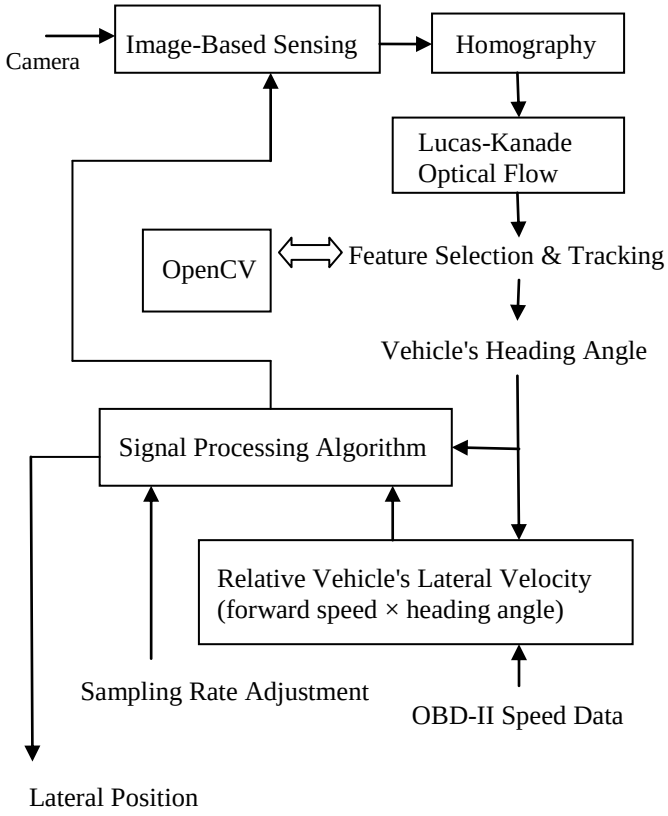

Fig. 9 Schematic diagram to determine the vehicle's lateral position.

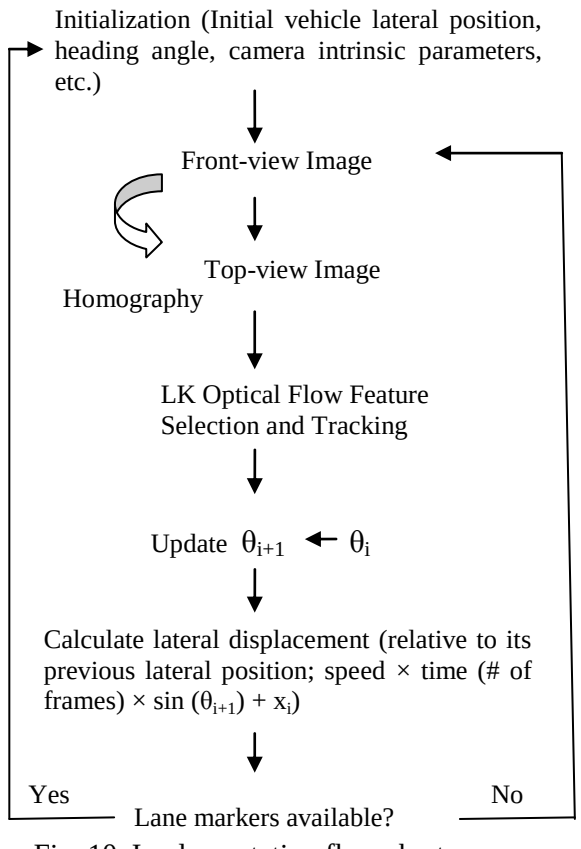

Fig. 10 Implementation flow chart.

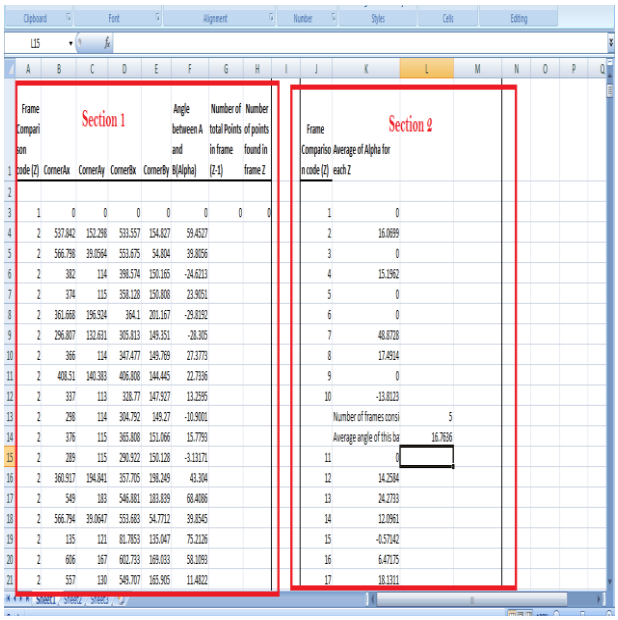

Fig. 11 A snap shot of the data spreadsheet.

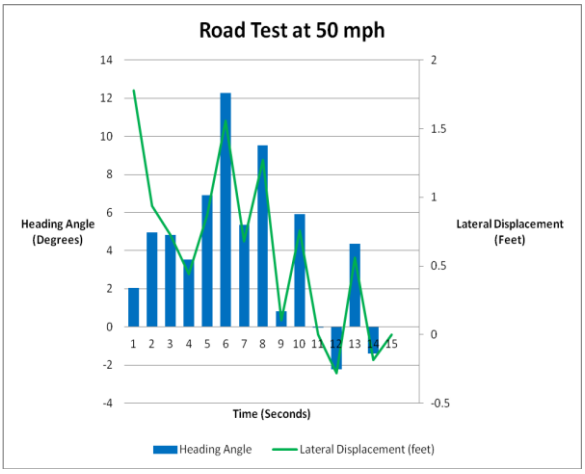

Fig. 12 Road test showing vehicle's heading angle and lateral position with 50 mph speed.

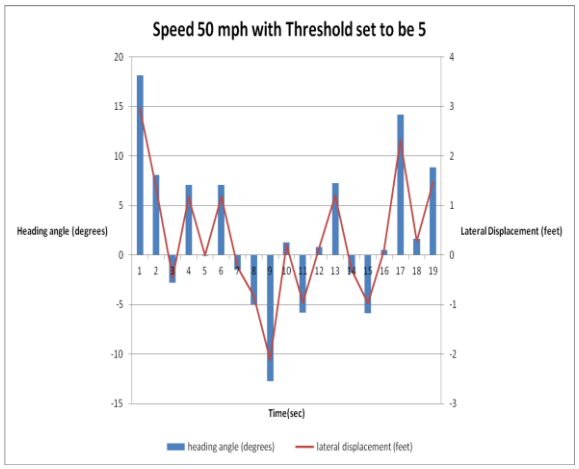

Fig. 13 The vehicle's heading angle and lateral position – Case 1.

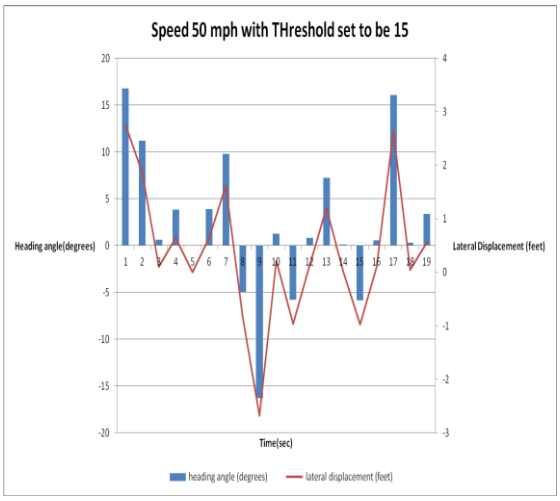

Fig. 14 The vehicle's heading angle and lateral position – Case 2.

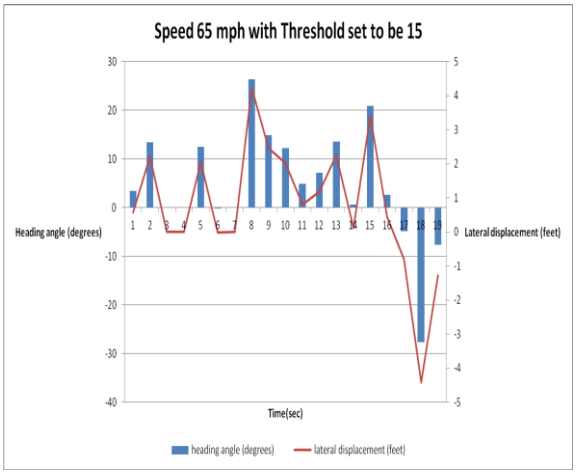

Fig. 15 The vehicle's heading angle and lateral position – Case 3.

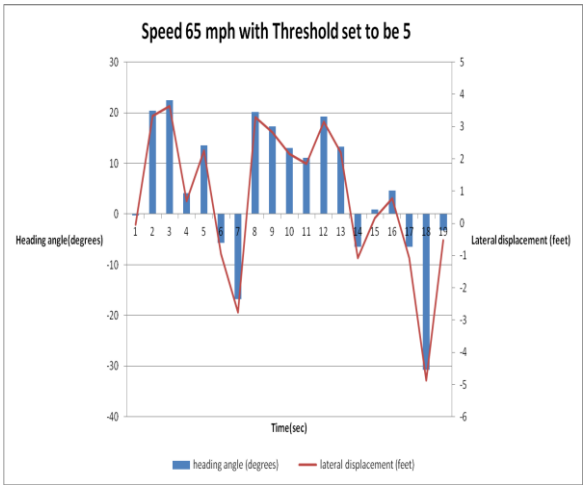

Fig. 16 The vehicle's heading angle and lateral position – Case 4.

Table 1 Averaged angle calculated from features

Frame Comparison Code (Z)	Average of Alpha for each Z (beta)	
1	0	
2	16.0699	
3	0	
4	15.1962	
5	-0.494116	
6	48.2741	
7	48.8728	
8	17.4914	
9	13.7032	
10	-13.8123	
Number of frames considered for taking average in this batch		8
Average angle of this batch		18.1626
11	-20.1296	
12	14.2584	
13	24.2733	
14	12.0961	
15	-0.5714	
16	6.4718	
17	18.1311	
18	26.9381	
19	-7.8043	
20	7.085	
Number of frames considered for taking average in this batch		10
Average angle of this batch		8.075

Table 2 A sample test data showing the heading angle and lateral displacement

Time (sec)	Heading angle (degrees)	Lateral Displacement (feet)
1	18.1626	2.97236
2	8.07484	1.33942
3	-2.77933	-0.462371
4	7.067	1.17315
5	-0.04038	-0.00672027
6	7.09555	1.17787
7	-1.51551	-0.25219
8	-4.963	-0.82494
9	-12.7655	-2.10698
10	1.28358	0.213603
11	-5.8179	-0.966587
12	0.787401	0.13104
13	7.23265	1.20051
14	-1.80271	-0.299968
15	-5.85942	-0.97346
16	0.541994	0.0902005
17	14.1561	2.33205
18	1.65034	0.274622