ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Accident Warning and Rest Recommendation System for Vehicles Using Gradient Vector Calculations.

A Saranya, S Suganthi, and V Vijayakumar*.

Department of Electronics and Communication Engineering, Sathyabama University, Chennai-600119, Tamil Nadu, India.

ABSTRACT

Driver distractions are the leading cause of most vehicle crashes and near crashes. According to studies, 80% of crashes and 65% of near-crashes involve some form of driver distraction. In addition, distractions typically occurred within three seconds before the vehicle crash. Recent reports have shown that the number of people injured in vehicle crashes related to distracted driving has increased. Many people were killed due to distracted driving crashes, which is a slight reduction. Distracted driving is defined as any activity that could divert a person's attention away from the primary task of driving. To provide solution for these problems driver inexpensive behavior monitoring device is proposed in this project.

Keywords: CCD camera, TTC, LINPACK, MATLAB

Corresponding author

INTRODUCTION

Driver fatigue is an important factor in a huge number of vehicle accidents. Recently, statistics estimate that per year 76,000 injuries and 1,200 deaths can be attributed to fatique related crashes. The enhancing of technology for preventing or detecting drowsiness at the wheel is a big challenge in the field of accident avoidance systems. The Volvo car company was first initiate and implements this system, known as Collision Avoidance System. In addition, this company provides audible and visual warnings and issues automatic steering and braking when it senses a similar collision with another person's or vehicle [1]. This kind of system involves to execute the high level algorithms in objective recognition and detection onto powerful and sophisticated custom made embedded systems. There are various preventable variables are involved in car accidents, such as impairment, fatigue and inattentiveness. Approximately, there are around 20-50 million injuries and 1.2 million deaths from road related accidents recorded around the world for every year caused by driver impairment, inattentiveness and fatique [2]. At present, more number of research going into driver assist technology which reduces accidents, such as tracking algorithms and lane recognition [3], speed limit displaying [4], lane keeping and lane tracking algorithms with the help of camera based image processing techniques by controlling the brakes and steering wheels [5].

Marius et al. [6] proposed a new technique of management and optimizing the highway traffic, based on the Constant Time to Collision Criterion. Sumio et al. [7] implemented a Lane Departure Warning System for car production with CCD camera and warn the driver with sound and visual messages, in addition steering wheel vibration. Eric et al. [8] reported the enhancement of a camera-based FCA technology which can also estimates TTC utilizing image scale change. Said et al. [9] proposed a Distance to Line Crossing (DLC) based time computation for line crossing with various computation methods. Lee et al. [10] demonstrated a collision avoidance and warning evaluation technique based on a metric performance which is commonly utilized for information retrieval and signal detection under unbalanced data population. The aim of this project is to develop a prototype drowsiness detection system. The focus will be placed on designing a system that will accurately monitor the open or closed state of the driver's eyes in real-time. By monitoring the eyes, it is believed that the symptoms of driver fatigue can be detected early enough to avoid a car accident. Detection of fatigue involves the observation of eye movements and blink patterns in a sequence of images of a face. Initially, we decided to go about detecting eye blink patterns using Matlab.

The procedure used was the geometric manipulation of intensity levels. The algorithm used was as follows. First we input the facial image using a webcam. Preprocessing was first performed by binarizing the image. The top and sides of the face were detected to narrow down the area where the eyes exist. Using the sides of the face, the center of the face was found which will be used as a reference when computing the left and right eyes. Moving down from the top of the face, horizontal averages of the face area were calculated. Large changes in the averages were used to define the eye area. There was little change in the horizontal average when the eyes were closed which was used to detect a blink. However Matlab had some serious limitations. The processing capacities required by Matlab were very high. Also there were some problems with speed in real time processing. Matlab was capable of processing only 4-5 frames per second. On a system with a low RAM this was even lower. As we all know an eye blink is a matter of milliseconds. Also a drivers head movements can be pretty fast.

Computational Details

The name MATLAB stands for MATrix LABoratory. MATLAB was written originally to provide easy access to matrix software developed by the LINPACK (linear system package) and EISPACK (Eigen system package) projects. MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research.

MATLAB has many advantages compared to conventional computer languages for solving technical problems. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. The software package has been commercially available since 1984 and is now considered as a standard tool at most universities and industries worldwide. It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. There are toolboxes for signal processing, symbolic computation, control theory, simulation, optimization, and several other Fields of applied science and engineering.

Image gradient

An image gradient is a directional change in the intensity or color in an image. Image gradients may be used to extract information from images. In graphics software for digital image editing, the term gradient or color gradient is used for a gradual blend of color which can be considered as an even gradation from low to high values, as used from white to black in the images to the right. Another name for this is color progression. Mathematically, the gradient of a two-variable function at each image point is a 2D vector with the components given by the derivatives in the horizontal and vertical directions. At each image point, the gradient vector points in the direction of largest possible intensity increase, and the length of the gradient vector corresponds to the rate of change in that direction. Since the intensity function of a digital image is only known at discrete points, derivatives of this function cannot be defined unless we assume that there is an underlying continuous intensity function which has been sampled at the image points. With some additional assumptions, the derivative of the continuous intensity function can be computed as a function on the sampled intensity function, i.e., the digital image. Approximations of these derivative functions can be defined at varying degrees of accuracy. The most common way to approximate the image gradient is to convolve an image with a kernel, such as the Sobel operator or Prewitt operator. The gradient of the image is one of the fundamental building blocks in image processing. For example the canny edge detector uses image gradient for edge detection. Image gradients are often utilized in maps and other visual representations of data in order to convey additional information. GIS tools use color progressions to indicate elevation and population density, among others.

Uses in Computer vision

Image gradients can be used to extract information from images. Gradient images are created from the original image for this purpose. Each pixel of a gradient image measures the change in intensity of that same point in the original image, in a given direction. To get the full range of direction, gradient images in the x and y directions are computed. One of the most common uses is in edge detection. After gradient images have been computed, pixels with large gradient values become possible edge pixels. The pixels with the largest gradient values in the direction of the gradient become edge pixels, and edges may be traced in the direction perpendicular to the gradient direction. One example of an edge detection algorithm that uses gradients is the canny edge detector. Image gradients can also be used for robust feature and texture matching. Different lighting or camera properties can cause two images of the same scene to have drastically different pixel values. This can cause matching algorithms to fail to match very similar or identical features. One way to solve this is to compute texture or feature signatures based on gradient images computed from the original images. These gradients are less susceptible to lighting and camera changes, so matching errors are reduced.

PROPOSED SYSTEM AND DESCRIPTION

This project proposes an inexpensive vision-based system to accurately detect Eyes off the Road (EOR). The system has three main components: 1) robust facial feature tracking; 2) head pose and gaze estimation. From the video stream of a camera installed on the steering wheel column, our system tracks facial features from the driver’s face. Using the tracked landmarks and a 3-D face model, the system computes head pose and gaze direction. The head pose estimation algorithm is robust to non rigid face deformations due to changes in expressions. Finally, using a 3-D geometric analysis, the system reliably detects EOR. The proposed system does not require any driver-dependent calibration or manual initialization and works in real time, during the day and night. To validate the performance of the system in a real car environment, we conducted a comprehensive experimental evaluation under a wide variety illumination conditions, facial expressions, and individuals.

ISSN: 0975-8585

Figure 1: Real Time Implementation

This section describes the main components of our system. There are six main modules: Image acquisition, facial feature detection and tracking, head pose estimation, gaze estimation, EOR detection, and sunglasses detection. A. Image Acquisition The image acquisition module is based on a low-cost CCD camera placed on top of the steering wheel column, The CCD camera was placed over the steering wheel column for two reasons: It facilitates the estimation of gaze angles, such as pitch, which is relevant for detecting when the driver is texting on a phone. From a production point of view, it is convenient to integrate a CCD camera into the dashboard. On the downside, when the wheel is turning there will be some frames in which the driver's face will be occluded by the steering wheel.

For night time operation, the system requires an illumination source to provide a clear image of the driver's face. Moreover, the illumination system cannot impact the driver's vision. To this end, an IR illuminator was installed on the car dashboard,. Note that the proposed system does not suffer from the common drawbacks of near-IR based systems, because it does not rely on the bright pupil effect. To adapt our CCD camera to IR illumination, it was necessary to remove the IR filter from the CCD camera, making the CCD more sensitive to IR illumination (i.e., sunlight, artificial illumination). As shown in Figure 1, this effect is not noticeable in real driving scenarios.

The driver's gaze direction provides crucial information as to whether the driver is distracted or not. Gaze estimation has been a long standing problem in computer vision. Most existing work follows a model-based approach to gaze estimation that assumes a 3D eye model, where the eye center is the origin of the gaze ray.

In this project, we used a similar model. We make three main assumptions:

- First, the eyeball is spherical and thus the eye center is at a fixed point (rigid point) relative to the head model;
- Second, all the eye points, including the pupil, are detected using the SDM tracker described in the previous section. Note that more accurate pupil center estimates are possible using other techniques such as the Hough transform;
- third, the eye is open and therefore all the eye contour points can be considered rigid.

Our algorithm has two main parts:

- Estimate the 3D position of the pupil from the rigid eye contour points
- estimate the 3D gaze direction from the pupil position and the eye center.

The 3D position of the pupil is computed as follows:

Triangulate the eye contour points in 2D and determine which triangle mesh contains the pupil.

Algorithm and Implementation

Drowsiness of a person can be measured by the extended period of time for which his/her eyes are in closed state. In our system, primary attention is given to the faster detection and processing of data. The number of frames for which eyes are closed is monitored. If the number of frames exceeds a certain value, then a warning message is generated on the display showing that the driver is feeling drowsy. In our algorithm, first the image is acquired by the webcam for processing. Then we use the Haar cascade file face.xml to search and detect the faces in each individual frame. If no face is detected then another frame is acquired. If a face is detected, then a region of interest in marked within the face. This region of interest contains the eyes. Defining a region of interest significantly reduces the computational requirements of the system.

After that the eyes are detected from the region of interest by using Haarcascade_eye.xml. If an eye is detected then there is no blink and the blink counter K is set to '0'. If the eyes are closed in a particular frame, then the blink counter is incremented and a blink is detected. When the eyes are closed for more than 4 frames then it is deducible that the driver is feeling drowsy. Hence drowsiness is detected and an alarm sounded. After that the whole process is repeated as long as the driver is driving the car. Figure 2 refers the flow chat of the proposed system

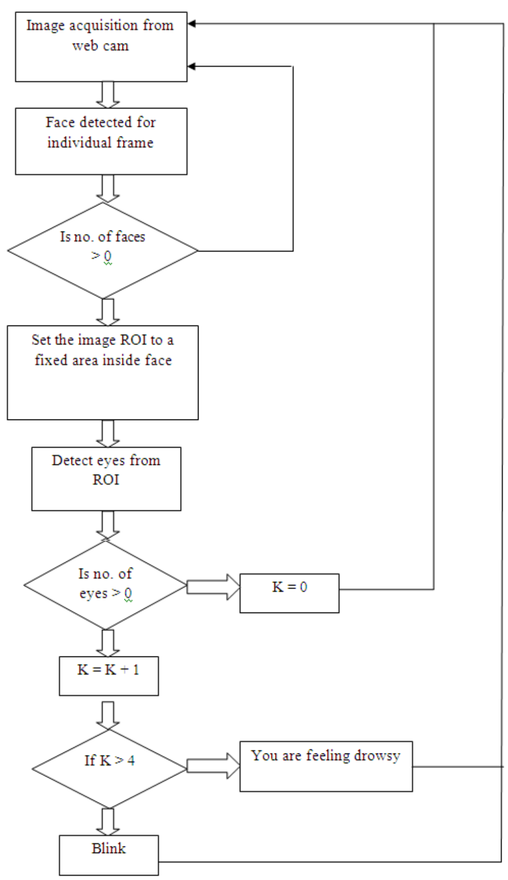

Figure 2: Flow chart

RESULTS AND DISCUSSIONS

To obtain the result a large number of videos were taken and their accuracy in determining eye blinks and drowsiness was tested. For this project we used a 5 megapixel USB webcam connected to the computer. The webcam had an inbuilt white LEDs attached to it for providing better illumination. In real time scenario, infrared LEDs should be used instead of white LEDs so that the system is non-intrusive. An external speaker is used to produce alert sound output in order to wake up the driver when drowsiness exceeds a certain threshold. The system was tested for different people in different ambient lighting conditions(daytime and nighttime). When the webcam backlight was turned ON and the face is kept at an optimum distance, then the system is able to detect blinks as well as drowsiness with more than 95% accuracy. This is a good result and can be implemented in real-time systems as well.

Sample outputs for various conditions in various images are given below. Two videos were taken; one in which only the eyes were detected and the other in which both face and eyes were detected.

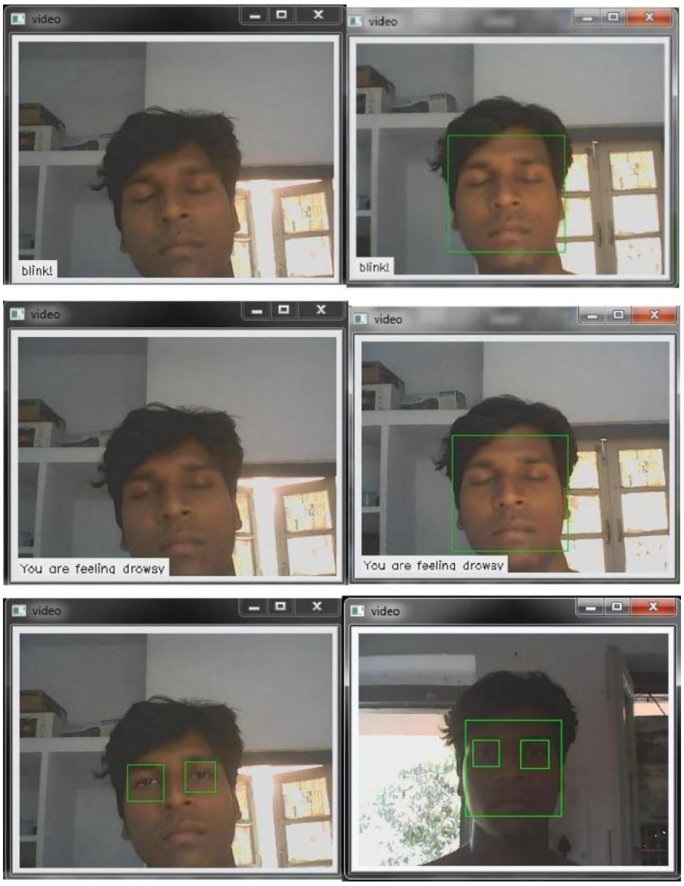

Figure 3: Face and eyes detected

Figure 3 represents the detection of face and eyes when the person who has feeling drowsy and blink eyes. The volunteer was asked to blink 20 times and become drowsy 6 times during the testing process. The accuracy for eye blink was calculated by the formula

Accuracy = 1 - |total no. of blinks -no. of blinks detected| / total no. of blinks.

The same formula was used for calculating accuracy of drowsiness detection.

It can be seen from the above table that if sample 5 is not taken into consideration then the system has an accuracy of nearly 100%. That said; the high amount of errors in sample 5 shows that the system is prone to error and has certain limitations which we will discuss in the next section. In sample 5 we did not use the backlight of the webcam. The resulting poor lighting conditions gave a highly erroneous output.

CONCLUSION

In conclusion, a prototype drowsiness detection system is successfully designed with the help of MATLAB tool. This system has been accurately monitored the open and closed state of the driver's eyes on real time. By monitoring the eyes, it is believed that the symptoms of driver fatigue can be detected early enough to avoid a car accident. Detection of fatigue involves the observation of eye movements and blink patterns in a sequence of images of a face. Thus, the prototype drowsiness detection is a most significant system which leads to avoid the unwanted car accident.

REFERENCES

[1] Jurnecka R. "Accident Avoidance: Volvo developing new safety systems". December 2007.
[2] World Health Organisation 2009, 'Global status report on road safety', Geneva, Switzerland.
[3] Petersson L, Apostoloff N, Zelinsky A, "Driver Assistance: An Integration of Vehicle Monitoring and Control", Proceedings of the 2003 IEEE International Con ference on Robotics and Automation 2003; 2:2097.
[4] Doshi A, Yuanhsien Cheng S, Manubhai Trivedi M. IEEE Transactions on systems, man, and cybernetics –Part B: Cybernetics 2009;39:39.
[5] Tuncer O, Guvenc L, Coskun F, Karsligil E, "Vision Based Lane Keeping Assistance Control Triggered By A Driver Inattention Monitor", IEEE International Conference on Systems Man and Cybernatics 2010;289.
[6] Balas, Marius M, The Traffic Management of Highways by Constant Time to Collision Cruise Control, Acta Technica Jaurinensis Series Logistica, 2009;2(3).
[7] Motoyama S, Ohta T, Watanabe T, Ito Y, Development of Lane Departure Warning System, 7th ITS World Congress 2000.
[8] Raphael E, Kiefer R. Int J Passeng Cars-Mech Syst 2011;4(1):467-478.
[9] Mammar S, Glaser S, Netto M, Time to Line Crossing for Lane Departure Avoidance: a Theoretical Study and an Experimental Setting, IEEE Transactions on Intelligent Transportation Systems 2006;7:226-241.
[10] Lee K, Peng H. Vehicle System Dynamics 2005;43: 735–751.