Intelligent Vehicle Braking System Using Image Processing

Goutam Prasad Sahay, Neeta Sahay, Suchismita Samanta

***Abstract*— The ever increasing accidents and injuries today call for those systems which will give safety and protection to the passengers in a running car and will ensure that high performance can be achieved and maintained under adverse conditions. To provide all these facilities, Intelligent Braking System is introduced in commercial vehicles providing rapid brake response and release for every single wheel of commercial vehicles according to the driver's demands. Implementation, here proposed, intelligent control strategy has be done by image processing technology based on object detection.**

***Index Terms*— Intelligent Braking System; Sixth Sense Technology; Computer Vision; Object Detection**

I. INTRODUCTION

Driving is a compulsory activity for most people. People use cars to move from one place to another. The number of vehicles is increasing day by day and in this way increasing the risk to accidents. Nowadays, the numbers of accidents are high and uncertain. Accidents occur frequently and cause worst damage, serious injuries and death. These accidents are mostly caused by delay of the driver to hit the brake. This project is designed to develop a new system that can solve this problem where drivers may not brake manually but the vehicles can stop automatically due to obstacles.

The main target for this project is cars can run automatic braking due to obstacles when the camera detects obstacles. Comparing the distance from the obstacles the car can control its speed and automatically apply brake according to the situation after receiving signal from the camera. The primary objective of this paper is to develop a safety car braking system using image processing and to design a vehicle with less human attention to the driving. While passive safety devices like seat belts and airbags help reduce injuries after a crash has occurred, active safety features like intelligent braking system, mandatory on all new passenger vehicles, are designed to help a driver avoid a crash. The project has been developed using MATLAB.

II. SYSTEM OVERVIEW

The implementation consists of three main components, that collectively acts as a system in itself and each device has its important role in the system. The devices include a high quality digital camera, the software to run the program and the programmed ECU controlling braking and accelerating system of the car. The camera is used to capture the objects in sight range and detect the obstacle in the road using object detection method and after various processing of the captured image the system will calculate the distance of the obstacle from the car. Then the distance is compared and according to the distance and the situation the car will control its speed and if needed it will apply the brake automatically.

III. SYSTEM DESCRIPTION

The work has been completed through different steps. The first step of the project is to capture the image with a webcam or acquire the image, i.e. Image Acquisition.

A. *Image Acquisition*

Image acquisition is the digitization and storage of an image. In order to process the image, the images should be acquired with the help of image acquisition hardware, a camera which act as a digital eye to the image processing system. The first stage of any vision system is the image acquisition stage. After the image has been obtained, various methods of processing can be applied to the image to perform the many different vision tasks required. Here MATLAB software is used as a software package for processing the data from the digital camera.

1) Creating Video Object: For a successful integration, a video object is created which helps to proceed with further processing of the data from the digital eye. Creation of video object is done in MATLAB with the help of integration of windows "winvideo" function with MATLAB. The following command helps to create a video object using

Video=videoinput (winvideo, 1) (1)

Here in the above command, Video is a variable name of the video object and the index number 1 represents the identity of camera. It can supports up to n numbers of input.

2) Setting Frames: The frames per second of the camera and other properties of the video object are also set. Frame grab interval is also set for better transition.

3) *Color Space Conversion:* Color space conversion is much needed for processing digital images. For applications regarding color recognition it's much better to use RGB color space format. RGB is a basic type of color image which comprises of three components namely R, G, and B

Manuscript received October 22, 2014.

Goutam Prasad Sahay, Computer Science and Engineering, Dr Sudhir Chnadra Sur Degree Engineering College, Kolkata, India
Suchismita Samanta, Computer Science and Engineering, Dr Sudhir Chnadra Sur Degree Engineering College, Kolkata, India
Neeta Sahay, Applied Electronics and Instrumentation Engineering, Academy of Technology, Hooghly, India

(grayscale type). Once the image has been converted into RGB image, then store it in a variable for further processing and feature extraction concepts to be applied.

4) Preview The Video Output: To view the video output in the user preferred size (window size) following command is used.

Preview (Video) (2)

5) *Snap the Video Data*: After a successful integration and preview of the video, the images from the real time video data are snapped. To get a snap of the present video data, “get snapshot” command is used along with the respective video object. A loop is set to get the acquisition frames continuously.

B. Object Detection

Detecting a object is a complex task. While recognizing a certain object in a scene, its background information is always trying to affect our region of interest. Another one issue is that, it always depends on the background illumination and luminance, chrominance of the scene. Upon overcoming these kinds of issue a perfect color recognized module is used with image subtraction method.

To detect the obstacle, MATLAB’s built in “imsubtract” function has been used. ”imsubtract” function can be used as,
Z = imsubtract (X, Y) (3)
Where, it subtracts each element in array Y from the corresponding element in array X and returns the difference in the corresponding element of the output array Z. X and Y are real, non-sparse numeric arrays of the same size and class, or Y is a double scalar. The array returned, Z, has the same size and class as X unless X is logical, in which case
Z is double. Here two consecutive frames captured by the camera is compared to detect any moving object. So with the help of ”imsubtract” function the first frame is subtracted from the second frame and the moving object in the second frame remains as the background is subtracted from the second frame. The image is converted to gray scale here. In this way obstacles are detected. The example is shown in Figure1.

Once extracted the object roughly, to give more intelligence to the system to redefine the size of the object which is extracted, morphological techniques like median filtration, binary conversion, removal of small unexpected pixels and like are used. These will redefine and enhance the size and shape of the object.

C. Filtering the Noise

After detecting the blue color in the input image, a median filter has been used to filter out the noise. Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges.

D. Converting Grey Scale Image into Binary Scale Image

To convert the gray scale image to binary scale image MATLAB’s built in “im2bw” function has been used. Function can be used as;

BW = im2bw (I, level) (4)

Where; it converts the grayscale image I to a binary image. The output image BW replaces all pixels in the input image with luminance greater than level with the value 1 (white) and replaces all other pixels with the value 0 (black). Specify level in the range [0, 1]. This range is relative to the signal levels possible for the image's class. Therefore, a level value of 0.5 is midway between black and white, regardless of class. In this study, the threshold 0.15-0.18 gave the best result for the large range of illumination change. The example is shown in Figure2.

E. Removing All the Small Areas

To get the best accurate number of the object detected in the image, all the areas other than the obstacle need to be removed. To do this, MATLAB’s “bwareaopen” function is used.
BW2 = bwareaopen(BW, P) (5)

Where; it removes from a binary image all connected components (objects) that have fewer than P pixels, producing another binary image, BW2. A threshold of 500 pixels for this study is set. The example is shown in Figure3.

F. Labeling

After removing all the connected components (objects) other than the obstacle, using MATLAB’s “bwlabel” function the obstacle can be labeled. In other words the region can be detected.

G. Feature Extraction

After recognizing the object some information about the recognized one needs to be extracted. To reduce the dimensionality of the object to give shape to the virtual object which originates in the physical world, feature extraction principles are used.

To get the features of the detected region such as center point or bounding box etc., MATLAB’s built in “regionprops” function can be used as;

STATS = regionprops (BW, properties) (6)

Where; it measures a set of properties for each connected component (object) in the binary image, BW. The image BW is a logical array; it can have any dimension. In this application the center of the object is used only.

H. Calculating the Distance

Using the feature extraction the center of the object is detected. Using the center of the particular object and the co-ordinate of the car the distance between the car and the obstacle is calculated. If the center of the object is (X1,Y1) and the co-ordinate of the center of the car is (X2,Y2), then the distance will be: $\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

[1] Controlling the Speed and Braking System

After calculating the distance between the car and the object this distance is used to control the braking system of the car. One particular threshold value is selected and stored previously in the system based on which the comparison of the distance will be done. Now the distance is compared with the threshold value. Now when the distance between the car and the object will decrease the speed of the car will also decrease and when the distance increases the speed of the car will increase and if the distance becomes equal to the threshold value then the car will apply the brake automatically to reduce the chance of collision.

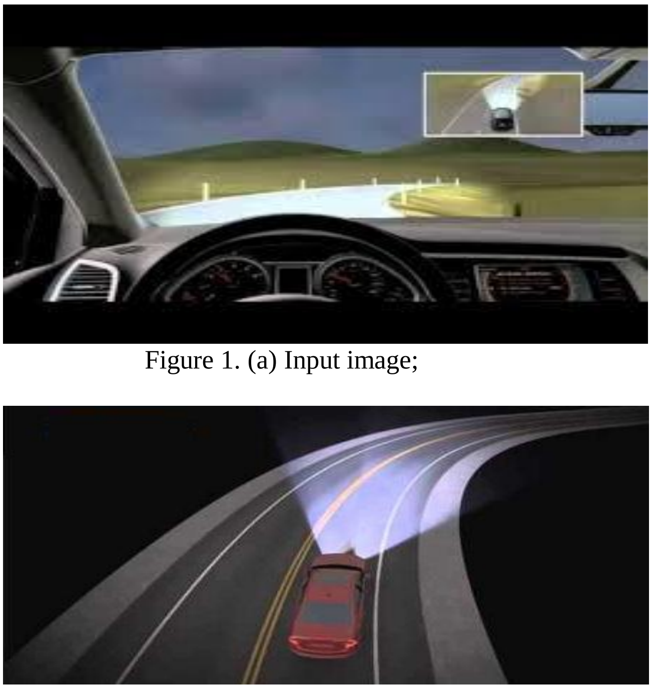

Figure 1. (a) Input image;

(b) Car following lane feeding images to the computer

Figure 2: Car reading the upcoming car's speed.

Figure 3: Car automatically slows down following the lane.

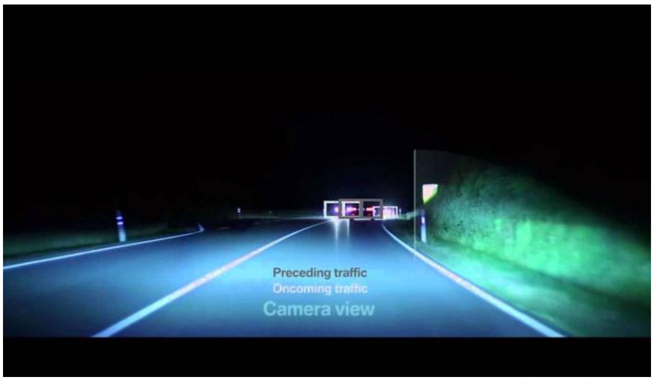

Figure 4: Camera reading oncoming traffic of both lanes.

IV. RESULTS

The output of this work is shown below. Here the center of the obstacle is detected first and then the distance between the car and the object is calculated. After that the distance is compared with a threshold value which is stored in the system previously. According to the comparison the speed and braking system of the car is controlled.

1. When a car or an object is coming towards the car in the same lane then the center of the object will be detected and the distance is also calculated. Now if the distance is decreased between the car and the moving object the speed of the car is automatically decreased as to maintain a safe distance to ensure the user safety and avoid injuries. Figure 1 and 2.

2. If the distance between the car and the moving object is increased the speed of the car will increase simultaneously with respect to the acceleration by the driver. If the distance between the car and the moving object and threshold value become same then there is a chance that head on collision can happen. That's why in that case the system will apply the brake automatically to avoid the crash. In this way this system will provide user safety and protection

IV. CONCLUTION

The use of image processing and object detection in MATLAB for the implementation of the proposed approach proved to be practically successful. The approach has huge potential once it gets further optimized, as its time complexity is higher, with the help of hardware having better specifications. This approach has much high potential for future advanced applications also. Also this would lead to a new era of Human Computer Interaction (HCI) where with the help of computer technology the safety of the passengers in a car can be ensured.

REFERENCES

[1] MI lanes, V. Onieva, E. Perez, J. De Pedro, T. Gonzalez, C. Control of Velocidad Adaptativo para Entornos Urbanos Congestionados. Rev. Iberoam, automat. Informat. Ind.2009, 66-73.

[2] Van NE's. N; Houtenbos. M; van SC Hagen. I: Improving Speed behavior: The Potential of In-Car Speed Assistance and Speed Limit Credibility. IET Intel. Transp. Syst. 2008, 2, 323-330.

[3] Joshua Pérez, Fernando Saco, Vicente MI lanes, Antonio Jiménez, Julio C. Diaz and Teresa de Pedro, an RFID based Intelligent Vehicle speed controller using active traffic signals, SENSORS 2010, 15872 5888; doi: 10.3390/s100605872.

[4] D. Aleksendrić , Č. Duboka, Ž. Arsenić, A Model of Coupling Force Control System, Proc. 33rd Int. Conf. "Meeting of Bus and Coach Experts", 2-4. September, Balaton, Hungary, 2002.

[5] EBS (EPB) Electronically Controlled Braking System – Description of System and Function, Copyright WABCO 1998.

[6] Aleksendrić D, Duboka C. Artificial technologies in sustainable braking system development, Int. J. Vehicle Design 2008; 46 (2): 237-249.

[7] A. L. Lilein, "Digital Signal Processors vs. Universal Microprocessors", ESIEE Paris, SPRA344, Sept. 1996

[8] M Atyabi, M S K Hosseini and M Mokhtari, "A webcam mouse: visual 3D tracking of body features to provide computer access for people with severe disabilities", Annual IEEE conference India, Pp. 1 – 6,2006.

[9] ojoon Park, "A Method For Controlling The Mouse Movement using a Real Time Camera", 2008, Brown University, Providence, RI ,USA, Department of computer science.

[10] A. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E. "Computer vision based mouse", Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASS). IEEE International Conference.

[11] Xu, and B. Yang., "Brake System Analysis, Reliability, Testing and Control using Bench Experiment. UCLA. UCB-ITS-PRR-97-10, 1997.

[12] J. K. Hedrick, J.C. Gerdes, D.B. Maciuca, and D. Swaroop. Brake System Modeling, Control and Integrated Brake/Throttle Switching: Phase 1. UCB. UCB-ITS-PRR-97-21, 1997.

[13] D. Popovic, and L. Vlacic, Mechatronics in Engineering Design and Product Development, Marcel Dekker, Inc. 1999.

[14] H. Reza, Z. Xu, and B. Yang, P. A. Ioannou, "Modeling and Control Design for a Computer-Controlled Brake System", IEEE Trans. On Control Systems Technology, Vol. 5, No. 3, May 1997.

[15] A New Spatial-Color Mean-Shift Object Tracking Algorithm with Scale and Orientation Estimation Chung-Wei Juan and Jwu-Sheng Hu, *Member, IEEE.*