Automatic Localization of Backward Collision of Vehicles Using a Camera

Anitha P., Gajesh K.R., Pruthviraj J., Santosh Kumar S., Nandini. C. and Bhaskara Rao

Department of Computer Science and Engineering,
Dayananda Sagar College of Engineering, Bangalore-78, India.
E-mail: anithacse88@gmail.com, gajeshramachandra@gmail.com, pruviraj@gmail.com, santosh.subhramani@gmail.com, laasyanandini@yahoo.com, bhaskararao.nadahalli@gmail.com

Abstract

Lots of rear end collisions due to driver inattention have been identified as a major automotive safety issue. A short advance warning can reduce the number and severity of the rear end collisions [4]. This paper describes how to avoid rear end collision when vehicles are moving in the reverse direction. In order to avoid many of the parking lot accidents, our system provides a Backward Collision Warning (BCW) - a new vehicle/obstacle detection method by calculating the area of the obstacle and comparing it to the stored threshold value. The images of the obstacles/objects are captured by monochrome vision camera when a car is moving in the reverse direction. The BCW system uses Digital Image Processing techniques to track the object at the rear end of the vehicle [2], Camera calibration is used to get the distance of the obstacle at the rear end. Kalman filters are used for tracking the obstacles. Secondly bounding box is used to bind(separate each objects by using a rectangular border based on the threshold) the objects. Region properties are used to estimate the area. After determining the area of the obstacle/object, TTC (Time to Collision) is calculated which triggers an alarm system that makes the driver attentive.

The proposed technique is tested on our own generated data sets on parking lot and busy roads. This methodology is found to be efficient and we are planning to test our proposed implementation on road for Real Time application.

Keywords: Backward Collision Warning (BCW), Rear End Collision, Time to Collision (TTC), Camera Calibration.

Introduction

Poor visibility and lack of attention is identified as one of the major reasons for rear end collisions in parking lots and busy roads. Hence the use of Backward Collision Warning (BCW) systems is of great importance. This provides one of the best solutions to avoid such accidents. Nowadays sophisticated cars contain radars and sensors to detect the objects in the rear. Many car manufacturers use the radar technology for finding the accurate range of the objects; however using a lot of radar and sensors is not an economical approach. The radars used do not exhibit better performance due to narrow field of view and poor lateral resolution. This has prevented such systems from entering into today's frequently changing market and demands. Fusion of the radar and vision technology is an attractive approach which can provide better performance, but this approach is costly.

Many of the research done on backward collision detection are not found to be high efficient in detecting and tracking the obstacles, henceforth lot of work can still be done in this area to provide the robust technique in overcoming the backward collision. Hence, we are moving in this direction to come out with a novel methodology which is cost effective and is of high efficiency.

Remarkable development in the field of both video and image processing has inspired many of the experts to work in these fields and to find many novel methods which make life easier and even save them.

Literature Survey

Jianzhu Cui et.al [5] has proposed a technique **"Vehicle Localization using a Single Camera" [2006]** to reduce the number and severity of the rear end collisions. This paper describes a Forward Collision Warning (FCW) system based on monocular vision, and presents a new vehicle detection method: appearance-based hypothesis generation, template tracking-based hypothesis verification which can remove false positive detections and automatic image matting for detection refinement. The FCW system uses time to collision (TTC)[1]to trigger the warning .In order to compute time to collision (TTC), firstly, haar and adaboost algorithm is utilized to detect the vehicle; Secondly, we use simplified Lucas-Kanade algorithm and virtual edge to remove false positive detection and use automatic image matting to do detection refinement; Thirdly, hierarchical tracking system is introduced for vehicle tracking; Camera calibration is utilized to get the headway distance and TTC at last. The use of a single low cost camera results in an affordable system which is simple to install. The FCW system has been tested in outdoor environment, showing robust and accurate performance.

"Mobileye - Advance Warning System" [2006] [1]: Mobileye AWS is an Advanced Warning System that can detect immediate forward collision danger and unintentional lane departure. With this functionality the system provides a timely warning for the most common causes of accidents in nowadays traffic. The 'smart' Mobileye monocular camera[1] analyses the upcoming road while driving the vehicle and registers objects like other vehicles and lane markings. Objects that form a potential danger are transmitted to the warning panel located inside the vehicle compartment, and warns the driver in advance for the upcoming danger.

Xuezhi Wen.et,.al.[2006] et.al, [8] has proposed a system called **“A Rear-Vehicle Detection System for Static Images Based on Monocular Vision” :**A rear-vehicle detection system of static images based on monocular vision is presented. It does not need the road boundary and lane information. Firstly, it segments the region of interest (ROI) by using the shadow underneath the vehicle and edges. Secondly, it accurately localizes the ROI by vehicle features such as symmetry, edges and the shadow underneath the vehicle, etc. Finally, it completes vehicle detection by combining knowledge-based and statistics-based methods [8]. Under various illuminations and different roads (different day time, different scenes: highway, urban common road, urban narrow road), the system shows good results of recognition and performance.

R. Okada et. al, [2003] [9] has proposed paper on **“Obstacle detection using projective invariant and vanishing lines”:** A method for detecting vehicles as obstacles in various road scenes using a single onboard camera. Vehicles are detected by testing whether the motion of a set of three horizontal line segments, which are always on the vehicles, satisfies the motion constraint of the ground plane or that of the surface plane of the vehicles [9]. The motion constraint of each plane is derived from the projective invariant combined with the vanishing line of the plane that is a prior knowledge of road scenes. The proposed method is implemented into a newly developed onboard LSI. Experimental results for real road scenes under various conditions show the effectiveness of the proposed method.

Carlo Tomasi Takeo Kanade [1991] [11] has proposed a White paper on **“Detection and Tracking of Point Features”:** The factorization method described in this series of reports requires an algorithm to track the motion of features in an image stream. Given the small inter-frame displacement made possible by the factorization approach, the best tracking method turns out to be the one proposed by Lucas and Kanade in 1981. The method defines the measure of match between fixed-size feature windows in the past and current frame as the sum of squared intensity differences over the windows.

The displacement is then defined as the one that minimizes this sum. For small motions, a linearization of the image intensities leads to a Newton-Raphson style minimization. In this report, after rederiving the method in a physically intuitive way, we answer the crucial question of how to choose the feature windows that are best suited for tracking.

Our selection criterion is based directly on the definition of the tracking algorithm, and expresses how well a feature can be tracked. As a result, the criterion is optimal by construction. We show by experiment that the performance of both the selection and the tracking algorithm are adequate for our factorization method, and we address the issue of how to detect occlusions. In the conclusion, we point out specific open questions for future research.

Peter Hillman et.al [12], Square Eyes Software has proposed a technique to calibrate the camera which is called **“Camera Calibration and Stereo Vision”** This white paper outlines a process for camera calibration: computing the mapping between points in the real world and where they arrive in the image. This allows graphics to be rendered into an image in the correct position. Given this information for a pair of stereo cameras, it is possible to reverse the process to compute the 3D position of a feature given its position in each image - one of the most important tasks in machine vision. The system presented here requires the capture of a calibration chart with known geometry.

LiangLi et.al [3] have proposed a paper on **License Plate Detection Method Using Vertical Boundary Pairs and Geometric Relationship**. License plate detection and recognition is a crucial and difficult issue for an ITS (Intelligent transportation System). This paper proposes a robust license plate detection method using vertical boundary pairs and geometric relationships. A robust and efficient approach to detecting license plates. The main disadvantage is that it is not satisfactory for some specific images due to bad illumination and practical situations.

Proposed Methodology

In this paper we propose Backward Collision Warning (BCW) system based on monocular vision. First we compute the background of the set of images. Based on these results the obstacle/object present at the rear is detected and tracked using the kalman filter [2], a very efficient way to detect objects in video of low resolution. Then the object detected by the kalman filter is fed into the image processing block for drawing a bounding box around the object. Once the bounding box is applied to the object, its area is determined using Regionprops. Based on the area a threshold distance is set which causes an alarm to trigger when the motion of the car is backwards and nearing the object/obstacle at the rear.

Backward Collision Warning (BCW) System is a novel method to detect and avoid backward collisions. It consists of the following steps as shown in Fig.1:

The system is mainly divided into four parts:

- Camera Calibration
- Object detection and Refinement.
- Detecting the Bounding Box with maximum area
- TTC.

Camera calibration

Camera calibration [9] is a crucial phase in most vision systems. We use camera calibration for computing TTC.

The equations used to derive the camera calibration are as follows:

C_h=PCRGw_h [9] represents a perspective transformation [10] involving two co-ordinate systems. We obtain the Cartesian coordinates (x, y) of the imaged point by the first and second components of c_h by the fourth. Converting the above equation to Cartesian coordinates we get:

$$x = \lambda \frac{(X - X0)cos\theta + (Y - Y0)sin\theta - r1}{-(X - X0)sin\theta sin\alpha + (Y - Y0)cos\theta sin\alpha - (Z - Z0)cos\alpha + r3 + \lambda} \quad (1)$$

$$y = \lambda \frac{-(X - X0)sin\theta cos\alpha + (Y - Y0)cos\theta cos\alpha + (Z - Z0)sin\alpha - r2}{-(X - X0)sin\theta sin\alpha + (Y - Y0)cos\theta sin\alpha - (Z - Z0)cos\alpha + r3 + \lambda} \quad (2)$$

With reference to A=PCRG. The elements of A contain all the camera parameters and $c_h = Aw_h$ [9]. Letting k=1 in the homogenous representation yields

$$\begin{bmatrix} c_{h1} \\ c_{h2} \\ c_{h3} \\ c_{h4} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3)$$

Substituting $c_{h1}=xc_{h4}$ and $c_{h2}=yc_{h4}$ in the above equation and the product yields:

$$\left.\begin{aligned} xc_{h4} &= a_{11}X+a_{12}Y+a_{13}Z+a_{14} \\ yc_{h4} &= a_{21}X+a_{22}Y+a_{23}Z+a_{24} \\ c_{h4} &= a_{41}X+a_{42}Y+a_{43}Z+a_{44} \end{aligned}\right\} \quad (4)$$

c_{h3} was ignored because it is related to z.

$$\left.\begin{aligned} a_{11}X+a_{12}Y+a_{13}Z-a_{41}xX-a_{42}xY-a_{43}xZ-a_{44}x+a_{14}=0 \\ a_{21}X+a_{22}Y+a_{23}Z-a_{41}yX-a_{42}yY-a_{43}yZ-a_{44}y+a_{24}=0 \end{aligned}\right\} \quad (5)$$

The calibration procedure then consists of the following steps

Step (a): obtaining m>=6 world points with known coordinates(X_i, Y_i, Z_i),i=1,2,…,m.

Step (b): imaging these points with the camera in a given position to obtain the corresponding image points (x_i,y_i),i=1,2,…,m.

Step (3): using these results in equations (5) we solve for the unknown coefficients.

We have successfully derived the camera calibration with accurate range values based on real time data samples that are collected using the camera which is suitable for system.

We have also taken some of the sample input data and worked towards obstacle tracking and detection. We have tested the proposed approach on many samples of data sets generated.

Kalman filter

Object Detection is done using the Kalman filter [2] which can be defined as the following:

Data fusion using a Kalman filter can assist computers to track objects in videos with low latency. The tracking of objects is a dynamic problem, using data from sensor and camera images that always suffer from noise. This can sometimes be reduced by using higher quality cameras and sensors but can never be eliminated, so it is often desirable to use a noise reduction method.

The iterative predictor-corrector nature of the Kalman filter can be helpful, because at each time instance only one constraint on the state variable need be considered. This process is repeated, considering a different constraint at every time instance. All the measured data are accumulated over time and help in predicting the state.

Video can also be pre-processed, perhaps using a segmentation technique, to reduce the computation and hence latency.

The Kalman filter is a recursive estimator. This means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state.

In this paper only the predictor part of the kalman filter is used for object detection, no correction is done to the predicted objects since the need is only to locate the object and not to track it.

Bounding box

Component labeling [9] gives each blob on the foreground image map a unique label, a list is produced with all of the labels in the binary image map and the corresponding number of pixels which have this label. The list is sorted in descending order of blob size. A bundle rectangle is drawn for all the detected objects and the largest rectangle amongst them is taken to be the object which is nearer to the vehicle.

Regionprops is used to measure the properties of image regions (blob analysis) which measures a set of properties for each labeled region L.

Time to Collision(TTC)

In our system, TTC [4] is used to trigger a warning to the driver regarding the object/obstacle at the rear end.

Erez Dagan [1] from Mobileye Company gave us the way to compute the TTC.

TTC is calculated using equation (7).

First we need to measure the rate of ‘optic inflation’.

If we define "optic inflation" to be “ds”

$$ds = dw/w \quad (6)$$

(‘w’ is the width of the target vehicle in the image, and ‘dw’ the difference of widths of that vehicle between 2 frames) than it is shown that

$$TTC = dt/ds \quad (7)$$

where, dt is the time gap between the 2 frames.

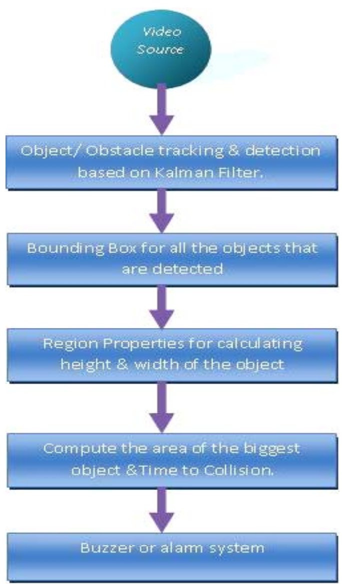

Fig.1: Steps followed in Backward Collision Warning (BCW) systems

As shown in fig.1 the Video captured is fed into the system from the camera at the rear end of the vehicle into the image processing System.

The System first slices the videos into multiple frames and then processes each frame accordingly. Firstly the camera calibration [9] is done to find the distance of the object from the vehicle. This distance will be used at the later stage to compute TTC [4] (Time to Collision). Based on the distance and the TTC an alarm is triggered to alert the driver about the obstacle at the rear end.

Secondly it uses Kalman filters [2] for tracking the object/obstacle which is at the rear end of the vehicle.

Bounding Box [7] is used to draw a rectangle around the objects detected in each frame.

The region props of the bounding box is used to find the area of the object, based on its height and width.

Sorting the areas in descending order of all the objects detected, will give the area of the biggest object in the top of the array. This is used to draw a bounding box with a red color. This is the biggest object in that particular frame. The rest of the objects are drawn with a bounding box of green color.

The buzzer or the alarm system alerts the driver when the car moves towards the object at the rear end. At a specific point using the area of the biggest object the system determines the threshold distance and alarms the driver when the car has reached the threshold distance. When the car reaches this distance from the obstacle, the alarm is triggered.

Experimental Results

In our system the threshold is taken to be 0.6 mtrs. The snapshots obtained during the experimentation for various conditions are shown in the figures below.

Images without Region of Interest (ROI)

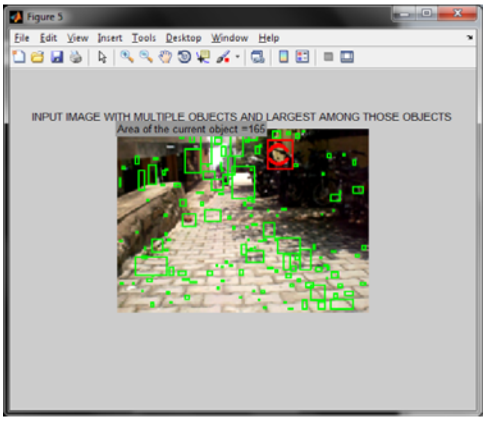

Fig2: Snapshot showing no Object or Region of Interest.

In fig.2 the vehicle is moving in the reverse direction and there are no objects or obstacles at the rear end. The red colored bounding box is the first biggest object that is detected, the other green bounding boxes are the objects that are small in area and are negligible.

Images where Region of interest are static

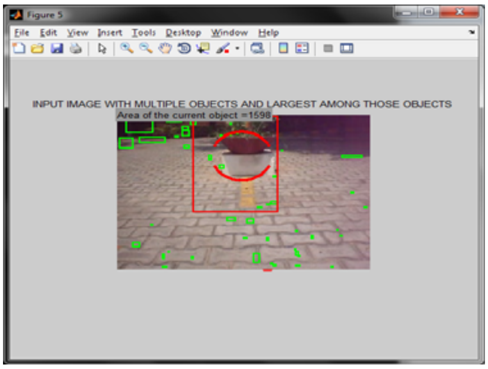

Fig3: sample largest bounding box extracted using region properties.

In the fig.3, the object or the obstacle is static in the ground. Hence the calculation of the area is a straight forward way. It is simpler to calculate TTC for such static objects.

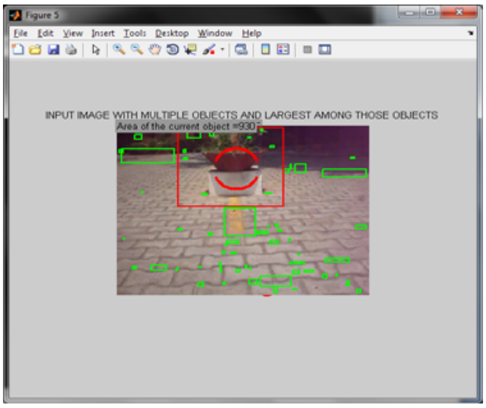

Fig 4: snapshot showing Vehicle approaching towards the object

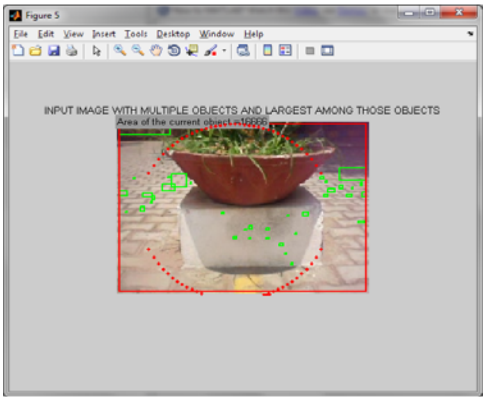

Fig 5: Snapshot showing Object is nearest to the car.

In fig.4 the vehicle is nearing the object in the rear and in fig.5 the car is too close to the object, at this point the threshold is crossed and the alarm buzzes.

Images where Region of interest are dynamic

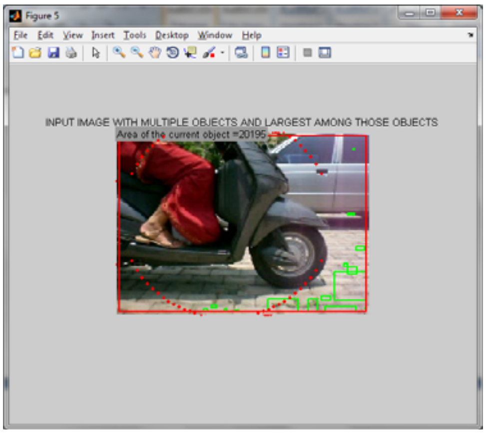

Fig.6: Snapshot showing the dynamic object moving in the rear end of the vehicle.

In test case fig.6, the dynamic object which is in motion is also detected.

Various kinds of datasets have been taken for checking the performance of the system in parking lots and even busy roads. The system works efficiently for various datasets and the results are obtained with accurate values.

.

Conclusion

The backward collision warning system can be implemented in all types of vehicles because of its cost effectiveness and affordable accessories which we collected the information in consultation with the automobile companies. Since most of the rear end driver related accidents is caused due to lack of attention by the driver[1], using this type of automatic obstacle detection system will not only assist the driver to safely move the car in the reverse direction, but also avoid the rear end collision.

The system can be implemented in all types of vehicles, both high cost and low cost and is found to be economically benefitted to the society. Since usage of high end radar systems are not affordable, our system not only helps in avoiding rear end collision but also in a cost effective manner, since the accessories used in the system are of low cost compared to that used in the existing backward collision warning systems (Eg. Range finders and optical sensors).

Henceforth a lot of work can still be done in this area to provide the robust technique in overcoming the backward collision. The methodology can be extended to take care of the orientation of the obstacles which are not in the perception of the camera.

References

[1] Mobil eye(Advance Warning System): A novel method of finding front end collision of vehicles in highways. http://www.mobileye.info/en/index.html

[2] The general descriptio about Kalman filter and its various types according to wikipedia site:http://en.wikipedia.org/wiki/Kalman_filter

[3] Meaning of collision detection according to wikipedia site:http://en.wikipedia.org/wiki/Collision_detection

[4] Jianzhu Cui,Fuqiang liu,Zhipeng Li,Zhen Jia, "Vehicle Localisation Using a Single Camera" 2010 IEEE Intelligent Vehicles Symposium, University of California, San Diego, CA, USA June 21-24, 2010

[5] E. Dagan, O. Mano, G.P. Stein and A. Shashua,"Forward Collision Warning with a Single Camera", IEEE Intelligent Vehicles Symposium (IV2004) June 2004, Parma, Italy

[6] Christopher Rasmussen, "Grouping Dominant Orientation for Ill- Structured Road Following," Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol.1, pp. I470-I477, 2004.

[7] Xuezhi Wen, Hong Zhao, Nan Wang, Huai Yuan, "A Rear-Vehicle Detection System for Static Images Based on Monocular Vision", School of Information Science & Engineering, Northeastern University Advanced Automotive Electronic Technology Research Center, Neusoft Co., Ltd. Shenyang, China, 2006 IEEE.

[8] R. Okada et al,"Obstacle detection using projective invariant and vanishing lines", Proceedings of the 9th ICCV 2003

[9] Digital image processing- Rafael C Gonzalez, Richard E. Woods ; ISBN: 0-201-43577-2.

[10] White Paper: Camera Calibration and Stereo Vision: Peter Hillman, Square Eyes Software 15/8 Lochrin Terrace, Edinburgh E-MAIL: pedro@peterhillman.org.uk, www.peterhillman.org.uk October 27, 2005