Asian Journal of Applied Science and Technology (AJAST)

Volume 1, Issue 5, Pages 14-16, June 2017

Detection of Static and Dynamic Obstacles for Self-Driving Cars

Farha Kowser[1] and Feby Jose Mathew[2]

[1]Assistant Professor, Department of ECE, MVJ College of Engineering, Bangalore, India. Email: farha.kouser@gmail.com
[2]UG Student, Department of ECE, MVJ College of Engineering, Bangalore, India. Email: feby.jose031995@gmail.com

Article Received: 12 May 2017 *Article Accepted: 01 June 2017* *Article Published: 04 June 2017*

ABSTRACT

Obstacle detection is one of the main features crucial for self-driving vehicles. To provide a higher degree of certainty, multiple detections are fused over time and between cameras to determine occupied and free spaces around the vehicle. Fish eye cameras are used which provides a wider view of the surroundings which is not in the case of conventional stereo cameras. Visual information is extracted as much as possible and combined through cognitive feedback. Camera position, depth maps and trajectories are solely based on such information. This can be especially useful in places like parking lots where the need to identify empty parking spaces without human intervention is highly necessary.

Keywords: Fisheye camera, Wheel odometry, Tracking, Disparities and Trajectories.

1. INTRODUCTION

Self-driving cars are an active field of research. To make it reliable it is vital for the car to sense both its static and dynamic environment. An occupancy map would suggest very little space for the car to move in Fig. 1 left picture which may be freed up in the next few moments as seen in the right picture. Obstacle detection approaches need to run in real time so that appropriate evasive measures can be taken. Active methods of obstacle detection make use of sensors such as laser scanners and ultrasound to identify obstacles. Passive make use of passive measurements like camera images which often work over a very wide range. Obstacles are seen as objects protruding from the ground. A wheel odometry system is used which determines poses of objects. Images from the cameras combined with depth map and wheel odometry is used as an input. Different couplings and feedbacks are used to support real time environment. The resulting detections are transferred into world coordinates and are grouped into candidate trajectories by the tracker. Successful detections are then again fed back for motion predictions [1-7].

2. EXISTING SYSTEM

The obstacle detection system mainly consists of four parts. In the first stage depth maps are constructed using a sequence of camera frames. The camera poses required for the step is obtained from the wheel odometry system and the extrinsic calibration of the cameras. In the second stage, detection and extraction of the obstacles and the free space is done. The appearance, depth and previous trajectories are also taken into account in the third stage. In the third stage, the information from the previous stage is fused with a static occupancy map. In the fourth stage, data is fused from several images captured by the camera in consecutive time intervals to provide accuracy [8-13]. Data can also be fused by capturing images from more than two cameras in the last stage. The entire system is causal which means that it depends on both the present and past inputs.

3. PROPOSED SYSTEM

The proposed system is to detect even dynamic obstacles. The output from the second stage along with the predictions of the tracker is used to update the pose estimation necessary to determine dynamic obstacles. Extended Kalman Filters are used for this purpose. The estimated trajectories are used to predict the future movements of the dynamic obstacles. The block diagram for all the steps is shown in Fig. 2.

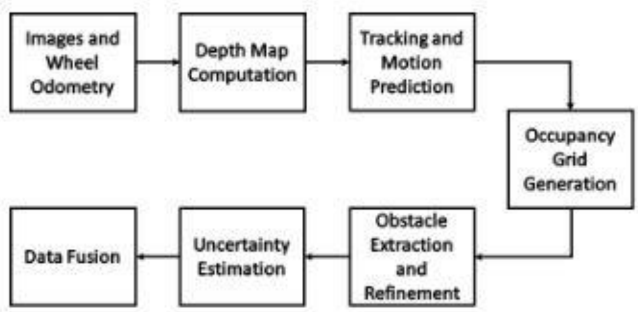

Fig.2: Block Diagram

4. DESCRIPTION

Following are the steps for obstacle detection:

4.1 IMAGES AND WHEEL ODOMETRY

The camera captures images of the scenes. Fish eye cameras give a wider scope in the view. The use of wheel odometry allows us to capture the poses of the camera. This is done by approximately determining the distance between the car and the obstacle and how much it differs when the car moves towards or away from the scene.

4.2 DEPTH MAP COMPUTATION

The second step is to formulate a depth map out of all the images and poses obtained. Images are captured in a continuous time rate. Normally the set of images are compared to a reference image. The images are projected onto a plane hypothesis and then to the reference image. If the depth of a pixel in the image is very close to that of the reference image, then the dissimilarity is less. In this paper, plane sweeping algorithm is used. Planes are swept in two directions. In one sweep direction, planes parallel to the ground plane is used and in the next sweep, planes fronto-parallel to the camera is used. A dissimilarity test is as mentioned above and a point per plane is obtained. To obtain a single value for each pixel, these points are averaged. Then both the sweeps are combined to form a depth map.

4.3 TRACKING AND MOTION PREDICTION

The detections of the current and past frames are accumulated in a space time volume. By starting the Extended Kalman filter on the detections at different time intervals, a set of trajectories is obtained which is used to predict the next motion poses. A set of algorithms in the EKF is used which provides automatic track initialization. The selected trajectories are used to provide a spatial before the next object detections.

4.4 OBSTACLE DETECTION

The detection consists of three steps: An occupancy grid is obtained which decide which cells contain the depth measurement. Obstacles are extracted from the occupied cells. Last, the uncertainty factor is estimated.

4.4.1 OCCUPANCY GRID GENERATION

This is to develop a two-dimensional array or grid of the occupancy points. The coordinate frame of the car coincides with the wheel odometry. The entire portion within the camera is represented as a Cartesian grid, each cell of equal sizes. This is to see which points are occupied and which are free, if free the car can move forward which are especially useful in parking lots. The coordinates in the occupancy map are expressed as (disparity, angle) pair where disparity is inverse depth value.

4.4.2 DYNAMIC OBSTACLES

The object selected by the tracker is modeled independently by EKF, we can predict its future position and obtain the corresponding uncertainty C. The object's dimensions will also be taken into consideration. This will enable us to determine free space while driving.

OBSTACLE EXTRACTION AND REFINEMENT

Obstacles need to be extracted from the occupancy grid. A ray is computed from each column of the grid corresponding to grid for which disparity values need to be calculated. A valid obstacle needs to have minimum depth measurement. There is a threshold value Tstrength to define obstacle strength. If below this value, then the object will not be taken as an obstacle.

4.4.3 UNCERTAINTY ESTIMATION

Since wheel odometry is used here, it is less accurate than the usually used visual odometry system. Therefore, an uncertainty factor needs to be taken into account. For fisheye cameras, the displacement error in pixels is linearly related to the angle difference between the viewing rays of 2 pixels. Intersection of 2 rays originating from the center of the camera, that creates an angle +/- delta uncertainty with an already existing ray from the center of the camera, provides an uncertainty level.

4.5 DATA FUSION

Assuming that the car moves on a planar ground, a 2D grid is again used to fuse obstacles from multiple camera frames. We follow a ray emerging from the center of the camera till it hits the first obstacle. A free space is given a negative weight while occupied space is given positive weight. The space behind the ray hitting the obstacle definitely has occupied space. Since the thickness of the obstacle is unknown a small region around the object is given a positive weight. After adding all the weights from the camera frames, the positive weights are considered as occupied spaces while the negative weights are considered as free spaces. The weights that sum up to give exactly zero are considered as unobserved. Weight added is determined by comparing the distance between the object and the camera with the distance between the cell and the camera. All the weights are assumed to be zero in the beginning and then the weights are updated with respect to wheel odometry.

5. RESULT

Thus, static and dynamic obstacles are successfully detected by the self-driving cars.

6. CONCLUSION

Thus in this paper, obstacles are detected using only fisheye cameras and wheel odometry. The basic feature is the grid representation which determines the occupied and free spaces. This grid can be fused with the outputs from other sensors too for better detection and analysis. Proper fusing from different sensors is also very challenging. The trajectory calculation can also be further improved by the use of GPS and LIDAR. This paper provides a detection security that is sufficient for applications running in real-time.

REFERENCES

[1] Y.-W. Seo, "Generating omni-directional view of neighboring objects for ensuring safe urban driving," Carnegie Mellon University, *Tech. Rep.*, 2014.

[2] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics, ser. Intelligent Robotics and Autonomous Agents. *The MIT Press*, 2005.

[3] Geiger, M. Lauer, and R. Urtasun, "A generative model for 3d urban scene understanding from movable platforms," *in Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[4] S. Thirthala and M. Pollefeys, "Radial multi-focal tensors," *International Journal of Computer Vision (IJCV)*, 2012.

[5] K. O. Arras, O. M. Mozos, and W. Burgard. Using boosted features for the detection of people in 2d range data. *In ICRA*, 2007.

[6] Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation*, 3(3):249–265, 1987.

[7] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, 73:41–59, 2007.

Volume 1, Issue 5, Pages 1-5, June 2017

[8] S. Nedevschi, R. Danescu, D. Frentiu, T. Graf, and R. Schmidt. High accuracy stereovision approach for obstacle detection on non-planar roads. *In Proc IEEE Intelligent Engineering Systems*, 2004.

[9] Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. *In IVS*, 2004.

[10] Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet part detectors. *IJCV*, 75(2):247–266, 2007.

[11] Christian Hanes; Torsten Sattler; Marc Pollefeys. Obstacle detection for self-driving cars using only monocular cameras and wheel odometry. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems.* Year: 2015. Pages: 5101 – 5108;

[12] A. Ess; B. Leibe; K. Schindler; L. Van Gool. Moving obstacle detection in highly dynamic scenes. *2009 IEEE International Conference on Robotics and Automation.* Year: 2009 Pages: 56 – 63.

[13] Mohammed, M.A., 2015. Design and Implementing an Efficient Expert Assistance System for car evaluation via fuzzy logic controller. *International Journal of Computer Science and Software Engineering (IJCSSE)*, 4(3), pp.60-8.