Real-Time Algorithm for Obstacle Avoidance Using a Stereoscopic Camera

Ioannis Kostavelis, Lazaros Nalpantidis and Antonios Gasteratos

Robotics and Automation Lab., Production and Management Engineering Dept.,
Democritus University of Thrace, University Campus, Kimmeria, GR-671 00 Xanthi, Greece

{ik3339,lanalpa,agaster}@pme.duth.gr

Abstract

This work presents a vision-based obstacle detection and avoidance method for autonomous mobile robots. The implementation of an algorithm able to navigate a robot in arbitrary environments usually demands of the synergy of several sensors. This work presents an algorithm employing only one sensor, i.e. a stereo camera, thus significantly diminishing the system's complexity. The implementation of this algorithm can be divided into two separate and independent modules. First, the stereo vision module retrieves information from the environment and produces disparity maps and then the decision making module analyses the data of the disparity maps and governs the robot's direction. The achieved frame rate ensures that the robot will have enough time to accomplish the proposed decisions in real time. Both of the modules have been implemented in C++. The complete algorithm has been examined by being applied on an extensive set of pre-captured stereo images.

Keywords: Obstacle avoidance, stereo vision, mobile robot navigation.

1. Introduction

In this work a vision-based obstacle avoidance algorithm is presented. This algorithm is intended to navigate a mobile robot. When a mobile robot moves in a real environment, its perception of the surrounding objects is crucial. This duty becomes harder due to the limited computational resources that a mobile platform usually offers. Taking into consideration that stereo vision is a computationally demanding process, the use of complex methods such as v-disparity can make the algorithm slow and inappropriate for real time navigation. Consequently, it is more reasonable to use simpler and more efficient methods. Mobile robots utilize stereo vision systems, as it is a reliable method to extract information about their environment [Iocchi et.al

(1998)]. Although stereo vision sensors provide an enormous amount of information, most of the mobile robots use secondary sensors in order to navigate safely [Siedwart et.al (2004)]. The proposed method comprises of two independent modules. The first one is the stereo vision algorithm. It is able to provide reliable depth maps, also known as disparity maps, of the scenery in frame rates suitable for a robot to move autonomously. The second module includes the decision making algorithm. It analyses the disparity maps and finds the most appropriate direction for the robot in order to avoid any possible obstacles. Both the modules have been implemented in C++. The complete algorithm has been designed to be flexible enough for use both in outdoor and indoor environments. Fig. 1 represents the flow chart of the implemented algorithm.

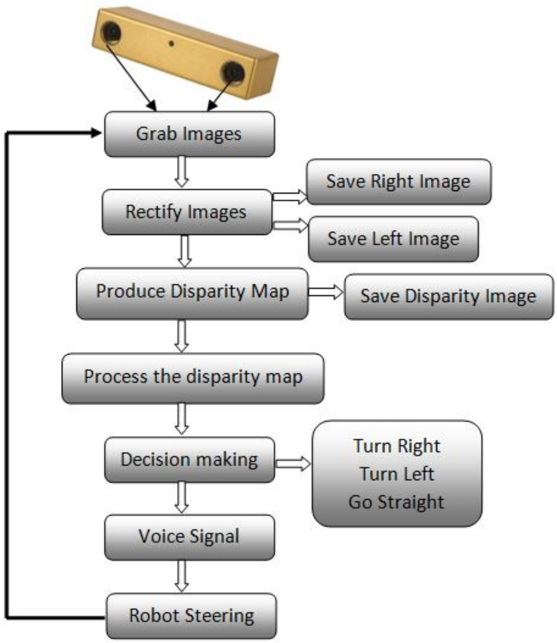

Figure 1. *Flow chart of the obstacle avoidance algorithm*

2. Related Work

In previous attempts for safe and undisturbed navigation, a wide range of sensors such as lasers and projectors have been used. Some important and enlightening references for sensor systems as well as obstacle detection and avoidance algorithms can be found in [Borrenstein et.al. 1989] and [Ohya et.al. 1998]. The majority of sensor systems can be divided into two categories. The first category involves the use of ultrasonic sensors. Their main advantages are the simple implementation and the ability to detect obstacles effectively. The second category involves vision-based sensor systems which can be further divided into two subgroups of sensor systems. The first one consists of stereo vision systems which are applied to the detection of 3D objects and the other one consists of laser range sensors which can be applied both in 2D and 3D obstacles but can be barely characterized by real-time operation [Vandorpe et.al, 1996]. Considering the above as a background the contribution of this work is the development of an algorithm for obstacle avoidance with the use of only one stereoscopic camera. The use of only one sensor and especially a stereoscopic camera diminish the complexity of our system and furthermore it can be easily integrated with other stereo vision modules such as object recognition and tracking ones.

3. Stereo Vision and depth estimation

The stereo vision equipment utilized in this work is the Bumblebee2 stereo camera by Point Grey Research. This is a packaged system that includes two pre-calibrated digital CCD cameras with 12cm baseline distance and a C/C++ software API for further custom stereo vision processing. The selection of this camera was based on the fact that Point Grey's stereo vision technology delivers full field of view depth measurements from a single image set. It also provides easy integration with other machine vision techniques.

3.1 The disparity map image

The disparity map is a very common and efficient method for storing the depth information of each pixel in an image. Firstly, images are captured by the stereo camera and are transmitted to a PC over the IEEE-1394 bus. Afterwards, they are corrected and aligned to remove any lens distortion. Carefully aligned and rectified stereo image pairs restrict the possibility of disparity existence only along the horizontal direction (i.e. there is no disparity along the vertical direction) for each pair of matching pixels. It is important to note that disparity is usually computed as a shift towards left of an image feature when viewed in the right image. A single point that appears at the horizontal coordinate x, in the left image may be present at the horizontal coordinate x-d in the right image, where by d is denoted the point's disparity in pixels. After rectification, a simple computational measure such as the sum of absolute differences (SAD) can be used to the find most similar pixels, and thus the disparity, for each pixel of the left image. There are stereo correspondence methods which calculate depth for every pixel of the scenery taking into consideration a small patch of pixels every time [Scharstein et.al. 2002]. On the other hand, there are methods that calculate correspondences accounting for the whole image. They are more accurate but at the same time more computational complex [Di Stefano et.al. 2004]. In a pair of images, objects that are closer to the capturing camera will exhibit greater displacement, i.e. greater disparity, and objects that are more distant will exhibit smaller displacement, i.e. smaller disparity. The right picture of Fig. 2 depicts the result of the stereo processing of a stereo image pair. The reference (left) image of this stereo pair is shown in left image of Fig. 2. The person who stands in front of Point Grey Bumblebee stereo camera system is closer to it and is readily visible in a light tone of grey. Objects that are in the background are marked with a darker tone of grey.

Figure 2. *The reference image (left) and the produced disparity map (right) for a stereo image pair.*

3.2 Point Grey disparity maps

The Point Grey system can produce disparity maps in 8 bit format which is an unsigned integer resulting in 256 levels of gray. Specific parameters of the produced disparity maps can be adjusted by dedicated Point Grey provided functions.

Firstly, a function is used in order to adjust the size of the correspondence window. This window is actually the stereo mask and the range of its values is between 1 and 25. A mask with value 1 means that the algorithm correlates each pixel in the image alone; whereas a mask with value 25 means that the algorithm is performing matching of 25x25 pixel patches. The result of this value’s selection has been examined in the scenery depicted in Fig. 3.

Figure 3. *Left and right images of a scenery that depicts obstacles in different depths*

For the needs of this work a mask with size of 15 pixels creates the optimum disparity image. The depth maps with smaller size of correspondence mask depict more details of the scenery but they also produces significant level of noise. On the other hand, large size of correspondence masks creates disparity images with less noise but depict fewer details, as shown in Fig. 4. Consequently, an intermediate value offers a balance between the two situations

Figure 4. *Detail preservation and noise occurrence in depth maps according to the value of the mask size (from left to right 1, 15, 25)*

Another parameter that should be adjusted is the maximum distance of the correspondence search. This value modulates the range of positions that the algorithm will search in order to match a window of pixels between two images. This parameter can take values between 2 and 255. Smaller values indicate that the algorithm will search among fewer pixels in order to find the correspondent pixel in the other image. In that way, only the objects that are at the background can be correctly matched. In order to match objects that are closer to the camera larger values should be used, as shown in Fig. 5. In our case, that is obstacle detection and avoidance, the objects that are closer to the robot are the crucial ones and must be avoided. As a result, the value used is 240 pixels.

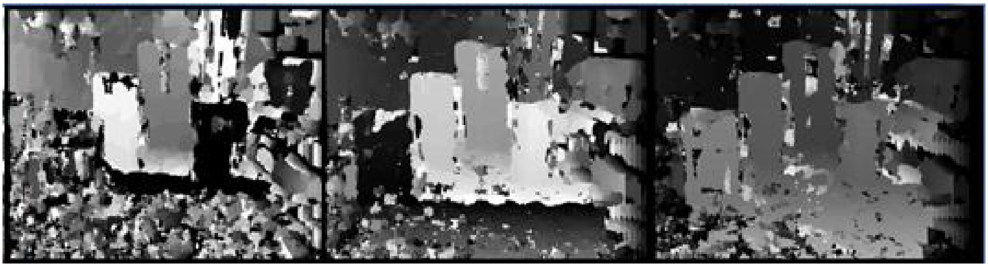

Figure 5. *Produced disparity maps for various values of maximum disparity (from left to right 50, 120, 240)*

4. Decision Making Methods and Algorithm Deployment

The calculated disparity map from the stereo camera's pair of frames is used in order to extract information for navigation. The goal of the developed algorithm is to detect obstacles and navigate the robot by steering it left, right or moving it forward. Contrary to many implementations that involve complex calculations upon the disparity map, the proposed decision making algorithm involves only simple summations and checks. Within this work two methods have been developed for obstacle avoidance, using the previously acquired disparity images, i.e. the mean estimation and the threshold estimation method.

4.1 The mean estimation method

The first method divides the disparity map into three windows of pixels. A predefined frame is subtracted in order to avoid mismatches due to different field of view of the two images. Then, the disparity map is divided into a left-side window, a central window and a right-side window, as shown in Fig.6. For each window, the pixels' disparity values are aggregated, as in Eq. 1. The aggregation results are then normalized considering the windows' dimensions, as in Eq. 2. Later on, the three normalized values are compared and the smaller one is selected. The corresponding window indicates the most preferable direction. The smaller value in one window indicates the smaller possibility of an obstacle existence in this window (e.g. if *median1<median2<median3*, the robot decides to steer left).

$$sum = \sum_{1} P_{ij} \qquad \textbf{(1)}$$

$$mean = \frac{sum}{num} \qquad \textbf{(2)}$$

Figure 6. *The separation into 3 windows.*

The *sum* is the aggregation result of the pixels' values in each window, the *mean* is the normalized aggregation result and the *num* is the number of pixels in each window. This method's implementation has exhibited problematic behavior during the evaluation phase. The problems come as a result of the nature of the calculations. In situations where the disparity image has a lot of noise the decision usually fails and the algorithm tends to steer the robot even if there are not any obstacles in front of the camera. This kind of behavior led to the development of the next, more sophisticated method.

4.2 The threshold estimation method

The second method also divides the disparity map into three windows of pixels, as shown in Fig. 6. In the central window, the pixels p whose disparity value $D(p)$ is greater than a defined threshold value T (i.e. T = 120), are enumerated. Then, the enumeration result is examined. If it is smaller than a predefined rate r (i.e. r = 20%) of all the central window's pixels, this means that there are not any obstacles and the robot can move forward. On the other hand if this enumeration exceeds the predefined rate, the algorithm examines the other two windows and chooses the one with the smaller average disparity value. In this way the window with the fewer obstacles will be selected. The values of parameters T and r play a significant role to the algorithm's decisions. In order to choose the optimum threshold value T and an appropriate rate value r o lot of experiments have been performed. Small values of T in conjunction with small values of r prevent the robot from moving forward, even when it should. For the purposes of this work the values T = 120 and r = 20% have been found to produce the best results in order to avoid obstacles given a disparity map with low level of noise. More specifically, in Fig. 6 the number of pixels of the central window whose value is greater than 120 is 40160 whereas the 20% of the pixels' count is 8160. As a result, the robot is not allowed move forward. The next step of the algorithm is the examination of the average disparity values between the two side windows. In this case, the left window has smaller average value than the right window (75.34<80.25) and consequently the final decision of the algorithm is to steer the robot left.

5. *Experimental validation*

The performance of the two proposed methodologies for obstacle avoidance has been examined by applying them on 25 disparity images. The experiments took place in outdoor scenes and the obstacles were natural elements like trees, bushes, benches and other randomly selected objects. The two algorithms had to detect and take the right decision in order to avoid as much as possible obstacles. All the 25 test image pairs are available to be freely downloaded from [Nalpantidis et.al. 2009]. Firstly, the performance of the two algorithms has been examined in order to conclude the most efficient method. The threshold method has spotted and avoided almost all the obstacles in the 25 pair of images, whereas the mean method took the right decision in less than the half situations, as indicated in Fig. 8.

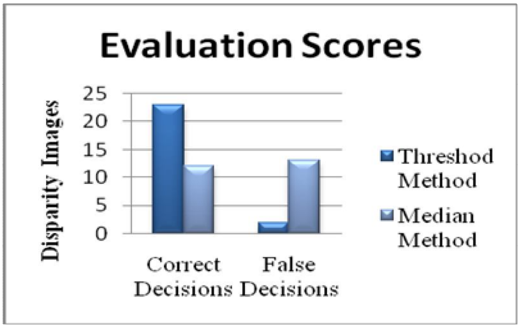

Figure 7. *The evaluation between the two methods*

Taking into consideration Fig. 7, only the threshold method is reliable for the avoidance of the obstacles. More specifically the median method seems to be unable to move the robot forward and in most cases it faulty decides to guide it left or right. This phenomenon has to do with the algorithms' structure. On the other hand, the threshold method managed to avoid the majority of the obstacles successfully.

In a second evaluation phase, the threshold method is examined more carefully. The algorithm spotted successfully all the right turns (100% success), the 83.33% of the left turns and the 93.33% of the moving forward decisions. These percentages reveal the reliability of the algorithms' decisions. Furthermore, the certainty of each decision is of interest as well. Due to the nature of the proposed algorithm, as presented in the previous sections, the decision about moving forward is based on some parameters' values and is totally independent from the other two possible decisions. Consequently, a measure of certainty would be meaningful only in the cases of the left or right decisions which can be directly compared. The certainty *cert* of a direction's decision which yields an average disparity avD_1 over the other direction witch yields $avD_2 > avD_1$ is calculated as:

$$cert = \frac{avD_2 - avD_1}{avD_2} \tag{3}$$

The results for the left and right decisions of the threshold method are shown in Fig. 8. For each decision the pair's indicating number as well as the algorithm's decision is given. The certainty ranges from 0% for no certainty at all, to 100% for absolute certainty. However, big values of certainty are not always achievable. In case that both left and right side are fully traversable the certainty measure would become 0%. Observing the correlation between false decisions and certainty values, a threshold could be decided, below which the algorithm should reconsider its decision.

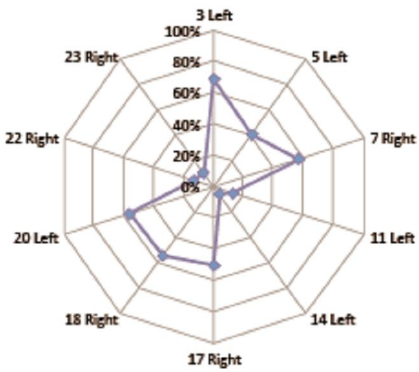

Figure 8. *Percentage of certainty for the threshold estimation method's decision*

6. Conclusions

In this paper a vision based algorithm for obstacle avoidance was presented. The sensorial needs of the proposed algorithm are restricted to a single stereoscopic camera and its computational burden is kept low. Two separate modules have been developed: one for stereo vision and another for navigation. Two methods were developed and examined for the obstacle avoidance module. The threshold estimation method has been found to be more efficient and reliable. Due to the good quality of the disparity images that are produced, the navigation methods are very effective. Thus, it is feasible to diminish the computational cost, as no Hough transformations of v-disparity calculations are required. The algorithm has been designed in such way that takes into consideration only the obstacles close to the camera.

The performance of the algorithm has been examined on self-captured outdoor images and its results have been evaluated. The threshold estimation method proved to be more reliable as it managed to avoid the obstacles successfully in the vast majority of the tested image pairs. The whole computation needs approximately 2.317 seconds to be completed. Extremely high performances are not of high significance because the robot should have enough time to physically implement the algorithm's decisions each time. Consequently, the proposed method is suitable for autonomous mobile robot navigation providing real time obstacle avoidance, based solely on a stereoscopic camera.

Acknowledgements

Mr. Nalpantidis is supported by the E.C. funded project View-Finder, FP6-IST-2005-045541.

References

1. Borrenstein J.,(1989), *Real time obstacle avoidance for fast mobile robots.* IEEE Trans Syst. Man Cybern, vol 19, pp.1179-1187.

2. Di Stefano L., Marchionni M., Mattoccia S. (2004)*, A fast area-based stereo matching algorithm.* Image and Vision Computing 22(12) 983-1005.

3. Iocchi L., Konolige K. (1998), *A Multiresolution Stereo Vision System for Mobile Robots.* Workshop on New Trends in Robotics, 1998.

4. Nalpantidis L., Kostavelis I. (2009), *http://robotics.pme.duth.gr/reposit/stereoroutes.zip* Group of Robotics and Cognitive Systems.

5. Ohya A., Kosaka A., Kak A., (1998), *Vision –based navigation by a mobile robot with obstacle avoidance using single camera-camera vision and ultrasonic sensing.* IEEE Trans. Robot. Automat. ,vol 14, pp.969-978.

6. Scharstein D., Szeliski R. (2002)*, A taxonomy and evaluation of dense two-frame stereocorrespondence algorithms.* International Journal of Computer Vision 47(1-3) 7-42.

7. Siedwart R., Nourbakhsh I. (2004), *Introduction to Autonomous Mobile Robots 2004.*

8. Vandorpe J., Van Brussel, H., Xu H., (1996), *Extraction Dynamic Map Building for a Mobile Robot using Geometrical Primitives Produced by a 2D Range Finder.* Proc. IEEE ICRA 96, vol.1, pp. 901-908.