Proceedings of ICAR 2003
The 11th International Conference on Advanced Robotics
Coimbra, Portugal, June 30 - July 3, 2003

A Real-Time Geometrical Obstacle Detection Algorithm For a Monocular Mobile Robot

Taha Kowsari, Majid Nili Ahmadabadi, Hamid Soltanian Zadeh

Control and Intelligent Processing Center of Excellence, Mobile Robot Lab., Department of Electrical & Computer Engineering
University of Tehran
kowsari@ece.ut.ac.ir, {mnili, hszadeh}@ut.ac.ir

Abstract

A real time monocular vision based obstacle detection algorithm is presented for static environments. The basic idea is developed from the observation that, the movements of the obstacle and of the floor are different in the image plane. This difference is proportional to the obstacle height.

In a calibrated vision system, tracking vectors can be directly converted to the motion field of the environment. Therefore, knowing the expected motion field of one of obstacle-free regions (bottom of the image for example) and assuming no obstacle in the view, the expected motion field of other regions is calculated. The obstacles can be detected easily by comparing the motion filed of each region in two consecutive frames with its expected value.

A geometrical model for the proposed idea is developed. Effects of different geometrical parameters and noise on minimum height of detectable obstacles are studies mathematical and experimentally. In addition, the proposed method is compared with divergence-based approaches.

The proposed algorithm is implemented on a mobile robot for real-time obstacle detection.

1 Introduction

Obstacle detection and avoidance are among the most important behaviors of any autonomous robot. Different types of sensors are used in the mobile robots for obstacle detection. Among them, vision system has received the most attention as it provides more information with a higher quality. But real-time and dependable obstacle detection methods yet have to be studied.

There are many researches on vision-based obstacle detection. A group of the studied methods extract 2D information from each single image [1][2][3]. In this approach a segmentation algorithm is used to separate the dangerous spaces from the safe background. For example Maja et al [1] have used an enhanced thresholding method for segmentation and implemented their method for real-time navigation.

The second group of researchers has used the 3D aspects of the obstacles in their obstacle detection methods. There are three main techniques to attain 3D information. In the first approach, additional sensors are used and their information is fused with the image data. Stereo vision systems implement this method [4]. Using a structured light to add some information to the images is the second technique [5]. Sequences of a single camera images are used in the third method to attain 3D information. The optical flow and the motion field-based methods are the most well known procedures in this category [6][7][8][9][10][11][12] [13][14][16].

In this paper a real-time optical flow based algorithm for obstacle detection with a calibrated monocular CCD camera is presented. The basic ideas of presented method and [16] are somehow similar however, the main idea of proposed method is inspired from physical observations. These observations enabled us to develop a geometrical model for the proposed method. Using this model, effects of different geometrical parameters and noise on minimum height of detectable obstacles are analyzed mathematically and experimentally. In addition, the presented method is compared with divergence-based approaches.

The assumptions, the basic idea and a mathematical model of the presented method are introduced in the next section. Then, the obstacle detection algorithm is presented. A simple navigation algorithm is introduced in section 4. Experimental results and effects of different geometrical parameters and noise on minimum height of detectable obstacles are discussed in the 5th section. Then the most related works are reviewed and a comparison is made. Last section gives the conclusion of this research.

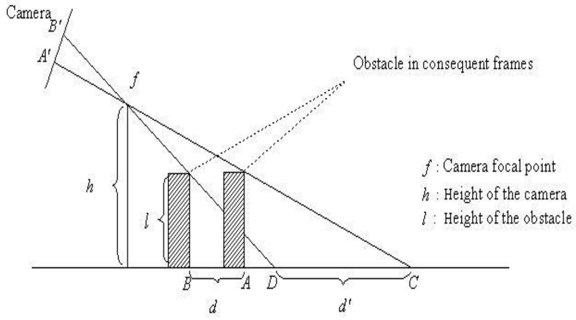


Figure 1: The camera is moved toward the obstacle. The displacement of the obstacle and the floor in the image plane are different. This difference is proportional to the height of the obstacle.

2 A Mathematical Model

Assumptions made in this work are very similar to those taken in [1] and many other researches. The first assumption is that, all obstacles rest on the ground and the environment is static. It is also supposed that the robot moves on a flat surface. In addition, it is assumed that the camera is tilted toward the ground.

Suppose that the camera is mounted on the mobile robot at a fixed height and tilt angle. The camera is directed toward the ground, see Figure 1. Assume that the mobile robot moves d units forward toward the obstacle located at A. This movement is equivalent to displacing the obstacle and the environment d units backward and toward the camera to point B. Doing so, the obstacle projection in the image plane moves from A' to B'. Projecting A' and B' back on the floor, points C and D are obtained.

The difference between d' and d depends on the height of the obstacle and of the camera from the ground. Mathematically speaking we have:

$$d' - d = \frac{l}{h-l} d \qquad \text{Eq.1}$$

where l is the height of the obstacle and h represents the distance of the camera from the ground. Exploiting the above relation is the core of the presented method.

There are different ways to take benefit from the above equation to detect the obstacles. Among those, we are interested in the simplest ones applicable in the real-time systems. The main steps of one of such methods are:

- Calibrating the camera so that we can project the image pixels to the corresponding points on the ground.
- Selecting some points in the image.
- Tracking the selected points and measuring their movements in the consequent frames using optical flow.

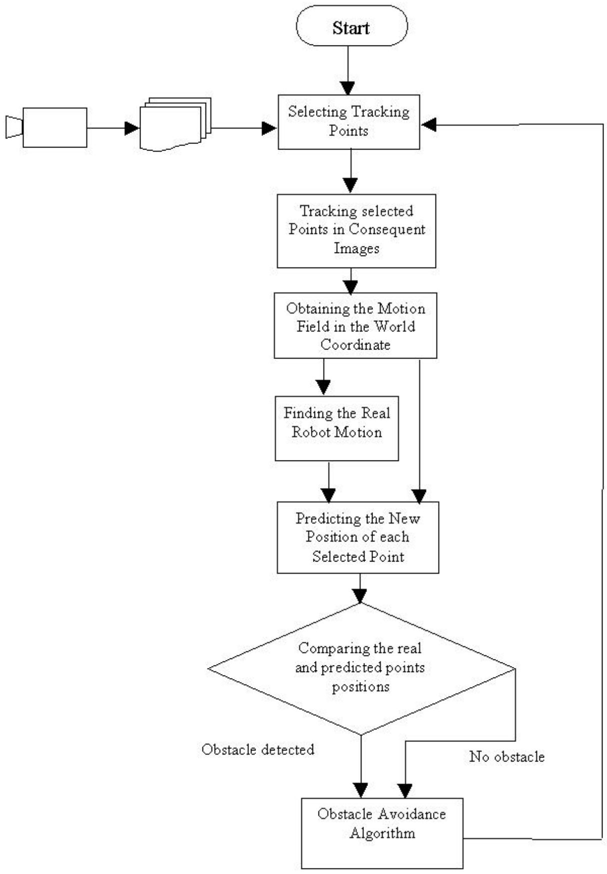


Figure 2: The flow chart of the proposed obstacle detection method.

- Finding the real translation and rotation of the camera in some ways and predicting the new position of the selected points assuming they are laid on the ground.
- Finding the difference of the selected point's predicted position with their new position obtained from optical flow. Points with non-zero difference are labeled obstacle.

3 Obstacle Detection Algorithm

Assume that the camera is fixed to the robot and is calibrated.

As shown in the flow chart in Figure 2, some points are selected for tracking first. In this research 225 points on a 15x15 grid are selected. Then, the optical flow of each selected point is calculated.

It is taken that the floor is seen at the bottom of the images. Therefore, the robot is able to calculate the camera rotation and translation from the optical flows of some points at the bottom of two consecutive frames[1].

[1]It is not a restricting assumption as the robot movements can be calculated by many other methods (ex. odometery).

After computing the rotation and translation of the camera and assuming that all selected points are on the ground, the positions of the other selected points in the new frame are predicted.

As mentioned, points on the obstacles have larger real optical flows, compared with those on the ground. Using this fact, the real and the predicted optical flows in the world coordinate are compared in the last step and points coming out of the floor are detected.

Because of noise in the image and presence of some weak-textured regions, some errors may exist in either the real optical flows or in the predicted positions. Therefore, a non-zero threshold must be used when comparing the expected and the real optical flows. This restricts the obstacle detection precision. Applying a temporal filter on the image may increase this precision.

Suppose that the robot moves forward d_w units and the projection of point A on the ground is displaced d_i units on the image plane, see Figure 3. We have

$$d_i = \frac{F}{h + d\cot\varphi} d_w \qquad \text{Eq.2}$$

Since the tracking precision is limited by fractions of pixel, according to Eq.1 and Eq.2, a threshold restricts the minimum height of detectable obstacles (we call this value MHDO hereafter). This threshold depends on the distance to the obstacle d, height of the camera h, the camera tilt angle φ, the focal length of the camera *F*, and the image resolution. Therefore, there is a trade off between the precision of obstacle detection and filed of view of the camera.

4 Navigation Algorithm

A simple obstacle avoidance algorithm is developed for testing the presented obstacle detection method. In this algorithm, the goal of robot is moving straight forward and avoiding the obstacles. At first, a danger value (*Dang)*, is computed as:

$$Dang = \sum_{k}^{k\ in\ obstacles} (i_{\max} - i_k)\alpha + (\frac{1}{2j_k - j_{\max}})\beta \qquad \text{Eq.3}$$

where *i* and *j* are respectively the number of row and column of considered obstacle point in the grid[2]. α and β are weights of contact danger in two directions. If |*Dang*| is more than a threshold, the robot turns left or right considering the sign of *Dang* value. The robot stops if |*Dang*| is greater than a second threshold. The experimental result of implementing the introduced obstacle detection and avoidance methods is

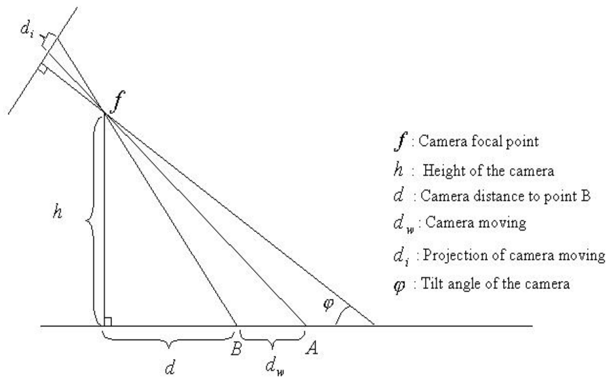


Figure 3: Movement in real world (from A to B) and its projection in the image plane.

presented in Figure 4-b. Figure 4-a shows the obstacles and the robot at its start point.

5 Experimental Results and Parameter Effects

The presented algorithm is tested on some real image sequences in real time for the mobile robot shown in Figure 4 (b). The computation cost of the presented algorithm for 225 points (a 15x15 grid) on a 384x288 pixel image is about 130 ms/frame on a 600 MHz Intel MMX CPU with 128 MB RAM under M.S. Windows 98. This means 8 frames/sec and is very acceptable for real time obstacle detection with this configuration.

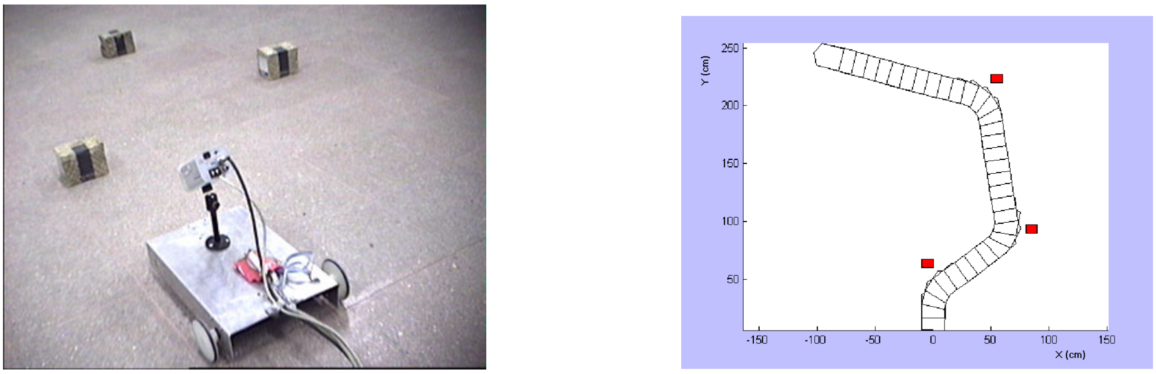
Figure 5 (a) depicts the difference between the real and the predicted positions of the tracked points in XY coordinates. Finally, in Figure 5 (b) the points detected in unsafe regions are shown.

In these experiments, the camera height and tilt angle are 23 cm and 40 degrees respectively. In this system, MHDO value is about 1.5 cm. For example in Figure 5 (b) there is a (10.5cm x 6.5 cm x 8cm) box and all of its selected points except some bottom points are marked as obstacle.

5.1 Parameters Effect

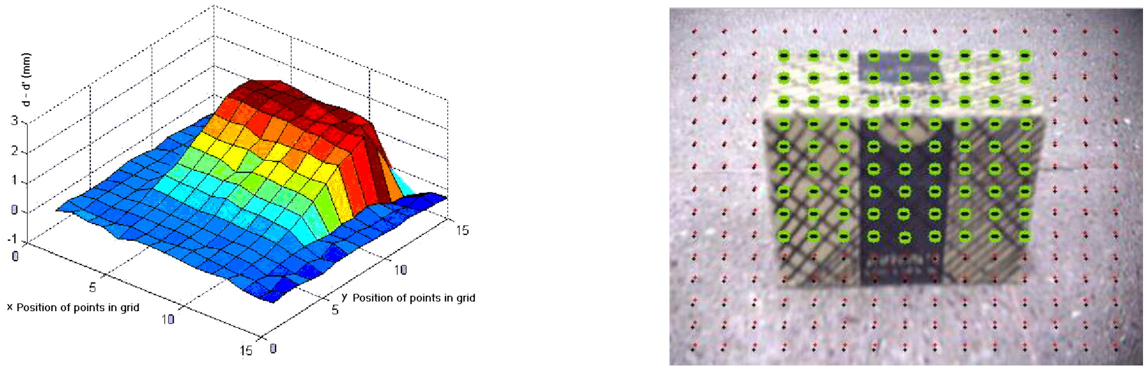
As mentioned in section 3, because of presence of noise and limitation on computation precision, MHDO is greater than a non-zero value. In fact, MHDO is a function of the camera configuration and motion and the obstacle distance in presence of noise and limited computation precision. In this section, effects of these parameters on MHDO are studied. It is clear that, more accurate optical flow is obtained for images with higher resolution. Therefore MHDO is in reverse proportion with the image resolution. In contrast, as the dimension of blocks used in optical flow computation is fixed, the computation time is increased for higher resolution images.

[2] The grid is on the robot coordinate frame.



(a) (b)

Figure 4: a) Robot and its environment clattered by obstacles. b) The path planed by obstacle avoidance algorithm.



(a) (b)

Figure 5: a) The experimental mobile robot. b. The computed optical flow vectors of the selected points. c) The differences between the real and the predicted motion field in the world coordinate. d) The obstacle is detected at the marked points

Figure 6-a shows $d'-d$ (the difference of real and predicted optical flow) for obstacles of heights 5 to 50 mm placed in different image depth. Applying the threshold, MHDO can be found. It is seen that MHDO increases for larger image depths.

According to equation Eq.1, $d'-d$ is in reverse proportion with the camera height. As the result, the signal to noise ratio is decreased for higher camera position. Consequently, MHDO is proportional with the camera height. Figure 6-b confirms this fact.

The robot speed affects MHDO value. The difference of two consecutive frames is increased for faster robot motion and fixed frame rate. Therefore, $d'-d$ and the optical flow computation time are increased. Consequently, MHDO is decreased for faster robot motions. Figure 6-c shows the experimental results.

Eq.2 shows that, propagation of a constant error in the optical flow and noise in the image affects d_w, and $d'-d$ consequently. According to this equation, a noise in the image affects $d'-d$ more for the smaller tilt angles (φ). Therefore, as shown in Figure 6-d, MHDO is larger for the smaller tilt angle of camera.

6 Related work

There are four main categories in the vision based obstacle detection algorithms. First category contains most of monocular segmentation based approaches. These approaches are computationally effective [1][2][3].Therefore, this set of algorithms could reach to a real-time implementation.

The main disadvantage of these methods is their 2D way of looking at the world. For example, in [1], Maja segmented a single image by his thresholding method to the safe and the dangerous regions. As reported in [1], his method fails when there is a shade or any pictures on the ground such as carpets etc. In fact, in these types of methods, there is no difference between the obstacles and their painted pictures on the ground.

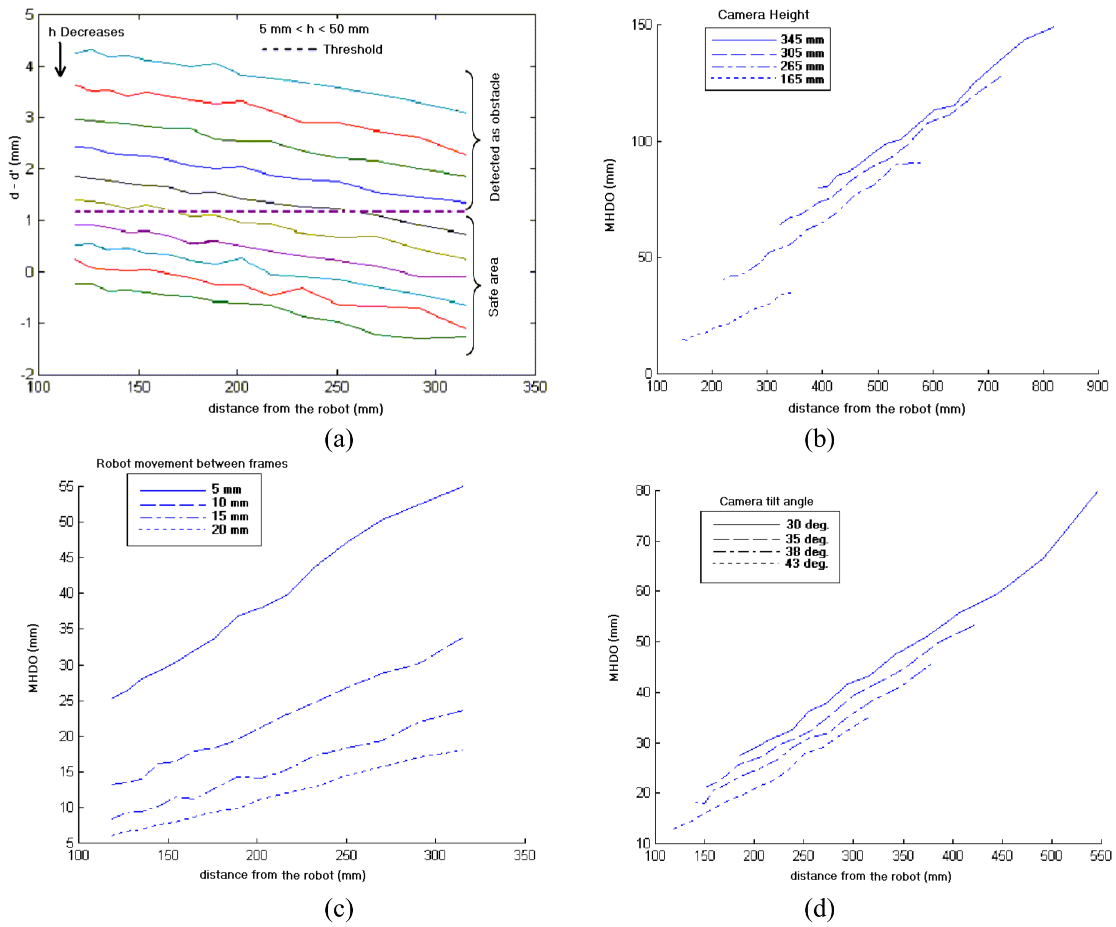


Figure 6: a) The difference of real and predicted motion field $d' - d$ for different heights of obstacles. b) MHDO value for different heights of the camera. c) MHDO value for different camera tilts angles. d) MHDO value for different robot movements between consecutive frames.

Second set of methods such as [4] , known as stereo vision based approaches, tries to achieve a 3D image by adding an extra camera. Having a 3D perception, these methods can recognize the obstacles more effectively. But these methods are computationally expensive. Also two cameras must be used that increases the system complexity, cost, and volume. The structured light based algorithms, such as [5], are categorized in the third group. These methods can obtain a 3D model of the world with less computation, compared with stereo vision systems. But the required equipment is expensive and they make artifacts in the environment, which are undesirable in most of the cases.

The last category consists of optical flow based algorithms. For example, Meyer [7] used a term known as TTC for detecting obstacles in some real images. Also Camus et al [12] used the divergence term for detecting obstacles in real-time and their robot could wander about 26 minutes without collision. Kruger et al [9] implemented a real-time system that uses an additional hardware for optical flow tracking.

In most of the reviewed researches, divergence is used as a criterion for obstacle detection. Divergence is independent of the camera rotation and is somehow proportional to the time to collision (TTC). As derivatives of the optical flow are required, there should be sufficient selected points in the image for the optical flow computation. It is not only time consuming but also the optical flows of a few points cannot be precisely computed. For instance, in weak-textured obstacles, only the edge points have valid optical flows. Our proposed algorithm exploits the natural effects of the obstacle geometry on the image and detects the obstacles by tracking each selected point separately. As a result, we can reduce our computation time by decreasing the number of the selected points, if necessary. This reduction affects the precision of the divergence method much.

In [16], a Kalaman filter is used on a series of consecutive image frames to find the difference of predicted and the calculated optical flow. The calculated difference is used for obstacle detection.

The obstacle detection method presented [16] is the most related research to our approach. In contrary to [16], the main idea of proposed method is inspired from physical observations. In addition, using these observations, we developed a geometrical model and studied the effects of different geometrical parameters and noise on minimum height of detectable obstacles.

7 Conclusion

In this paper, a simple and computationally fast obstacle detection method is proposed and effects of different geometrical parameters and noise on minimum height of detectable obstacles are studied.

It is analytically discussed that, the presented method exploits the effects of object's geometrical aspects on the optical flow. Therefore, it is possible to increase the minimum height of detectable obstacles by changing some parameters such as tilt angle and height of the camera and the robot speed.

The developed method can be used more efficiently relative to divergence-based algorithms. The implementation on a PC-based system showed that it could be used in unknown stationary environment for real-time obstacle avoidance.

Results indicated that, the presented method is much less sensitive to noise when compared with divergent-based methods. In addition, as there is no need for computing the derivative of the obtained optical flow (divergence), we can easily select only the points located in relatively strong-textured regions. This is another advantage of our approach over the divergence-based algorithms.

Using color information for increasing robustness of the proposed method is the next step of this research.

8 References

[1] JM. Maja T.Takahashi Z.D. Wang E.Nakano , "Real-Time Obstacle Avoidance for Visual Navigation", Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000).

[2] C. Bidlack, A. Hampapur, A. Katakere , "Visual Robot Navigation using Flat Earth Obstacle Projection", Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego , May 1994.

[3] M.T. Chao, T. Braunl and A. Zaknich , "Visually-Guided Obstacle Avoidance in office environments."Proceedings of the 6th International Conference on Neural Information Processing, Perth, Australia, 16-20th November, 1999, pp. 650-655.

[4] Y. Matsumoto , T. Shibata , K. sakai, M.Inaba, H. Inoue,"Real-time Color Stereo Vision System for a Mobile Robot based on Field Multiplexing" , Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque , New Mexico - Apr 1997, pp. 1934-1939

[5] J. J. L. Moigne and A. M. Waxman, Structured Light Patterns for Robot Mobility, IEEE Journal of Robotics and Automation, Vol.4, No.5, October 1988, pp. 541-548

[6] G. Baratoff , C. Toepfer, M. Wende,H. Neumann,"Real-Time Navigation and Obstacle Avoidance from Optical flow on a Space-Variant Map, Proc. of IEEE ISIC/Cicra/ISAS joint Conference , Sep. 1998, pp. 289-294.

[7] F. G. Meyer , "Time-to-Collision from First-Order Models of the Motion Field" , IEEE Transactions on Robotics and Automation, Vol. 10, No 6, December 1994, pages 792-798.

[8] A. Yakovleff, D. Abbott, X.T Nguyen, and K. Eshraghian, "Obstacle Avoidance and Motion Induced Navigation," in Proc. International Workshop on Computer Architecture for Machine Perception, Como, Italy, 18-20 Sept. 1995, pp. 384-393.

[9] W. Kruger, W. Enkelmann, and S. Rossle, "*Real-time estimation and tracking of optical flow vectors for obstacle detection*," in Proc. IEEE Intelligent Vehicles '95, Detroit, MI, pp. 304-309.

[10] R. C. Nelson, J. Aloimonos , "Obstacle Avoidance Using Flow Field Divergence", IEEE Transactions on PAMI, Vol. 11, 10, Oct. 1989, 1102-1106

[11] D. Coombs , M. Herman, T. Hong, M. Nashman, "Real-Time Obstacle Avoidance Using Central Flow Divergence , and Peripheral Flow", IEEE Trans. on robotics and Automation, Vol. 14, No. 1, Feb. 1998, pp. 49-59.

[12] T. Camus , D. Coombs , M. Herman, T. Hong, M. Nashman, "Real-Time Single-Workstation Obstacle Avoidance Using Only Wide-Field Flow Divergence", Journal of computer vision research, Vol. 1 , No. 3, MIT press, 1999.

[13] Kamal K. Gupta and Yong Yu,"On Eye-Sensor based Path Planning for Robots with non-trivial Geometry/Kinematics", In Proceedings of the 2001 IEEE International Conference on Robotics & Automation Seoul, Korea, May 21-26, 2001, pp. 265-270.

[14] Kai-Tai Song and Jui-Hsiang Huang,"Fast Optical Flow Estimation and Its Application to Real-time Obstacle Avoidance", In Proceedings of the 2001 IEEE International Conference on Robotics & Automation Seoul, Korea, May 21-26, 2001, pp. 2891-2896.

[15] Intel Computer Vision Library Documentation, http://sourceforge.net/projects/opencvlibrary/. last visited on September 15, 2002.

[16] W. Kruger, W. Enkelmann, and S. Rossle, "Real-time estimation and tracking of optical flow vectors for obstacle detection," in Proc. IEEE Intelligent Vehicles '95, Detroit, MI, pp. 304--309.