APEM journal Advances in Production Engineering & Management 2 (2007) 2, 79-90
ISSN 1854-6250 Original scientific paper

OBSTACLE DETECTION WITH ACTIVE LASER TRIANGULATION

Klančnik, S.[*]; Balič, J.[*] & Planinšič, P.[**]
University of Maribor
[*]Faculty of Mechanical Engineering, Smetanova 17, 2000 Maribor, Slovenia,
[**]Faculty of Electrical Engineering and Computer Science, Smetanova 17, 2000 Maribor
E-Mail: simon.klancnik@uni-mb.si, joze.balic@uni-mb.si, peter.planinsic@uni-mb.si

Abstract:
In this article we present the use of an obstacle-detection system that functions with the help of laser triangulation. We have developed this system to increase the safety of speech-controlled wheelchair. The given position of the set-up camera and laser, and the information regarding the position of a projected laser dot (line, grid, structure), which is acquired with the proposed calculation-efficient methods of image processing, enable us to determine the presence and position of the obstacle in the course. Furthermore, we propose an upgrade of the system by using fuzzy edge detection of object edges, which gives us additional information about the obstacle. Test results showed efficiency of the proposed solution to obstacle detection.

Key Words: Laser Triangulation, Image Processing, Fuzzy Edge Detection, Obstacle Detection

1. INTRODUCTION

In the recent years there has been an increase in the usage of image processing applications used in everyday life. These applications are capable of detecting, removing or isolating the important data obtained from the images.

For the past few years at our faculty, we have been actively developing an autonomous wheelchair VOIC, which can be operated by voice commands [1]. The wheelchair is designed primarily for tetraplegics, people paralyzed below the neck area and people who suffer by different forms of cerebral palsy.

Until the present, we have designed a system for voice recognition by the feed-forward neural network. But since the control of the wheelchair by voice commands is not entirely reliable, we have also developed an additional safety system which detects different obstacles with ultrasound sensors and thus prevents the collision of the wheelchair with an obstacle.

Our main focus has been on the problem of safe operating and driving of the wheelchair, and in this purpose we have suggested a solution of obstacle detection with active laser triangulation. By employing this system we wish to supplement the existing ultrasound sensors and increase the level of safety. We will consider all the objects that can not be driven over in the direction of the drive as obstacles.

System for obstacle detection with the usage of laser triangulation also has a wide range of potential usefulness in all the areas where information about the presence and position of obstacles in the environment is necessary (mobile robots, systems for car parking...). With the usage of a projected laser line, the system can also be used for scanning objects and detecting flaws in products during a production process.

Obstacles in images are usually detected with methods of edge detection and are therefore recognized in the image as edges. Numerous methods such as the Canny algorithm [2, 3, 10], the Sobel method [2, 3, 10], the Prewitt algorithm [2, 3, 10] and the fuzzy logic approach [4] to edge detection are being used in this aspect. These methods detect optical edges which represent the edges of an obstacle in the image, but they do not provide the depth information about the obstacle (the distance at which the obstacle is located is not known). For determining the depth information (third dimension) from 2D images captured by the camera, methods such as structured light and stereo vision are being used. The latter is the most common and it is based on merging two or more images captured from different positions. Two or more cameras are involved in this process. This system is basically very similar to the functioning of a human eye.

In this paper we will present the method of obstacle detection with the help of active laser triangulation [5]. We will describe in detail the algorithm used for detecting obstacles with a projected laser dot. The algorithm can easily be expanded with additional laser dots or a line, by which we increase the area of obstacle detection.

Since in this way we only obtain information about the area illuminated by laser light, we will also present one of the possible expansions of the system by employing a detection of optical edges with the fuzzy edge detection method. This provides us with additional information about the obstacle.

The article is organized in the following order. In the second chapter we will present the hardware and software used in the realization of the project. In the third chapter the principle of active laser triangulation and its functioning are described. The fourth chapter presents the method and the process of obtaining the needed information from an image. In the fifth chapter, the fuzzy edge detection method, which serves as a possible expansion of the system, is presented. The article ends with the presentation of results and a conclusion.

2. HARDWARE AND SOFTWARE EQUIPMENT

The application described in the present paper requires capturing an image of the surroundings, from which the information about eventual obstacles is then obtained. For image capturing, we used Sony DCR-HC23 digital camera [6] with 20x optical and 640x digital zoom.

Figure 1: Laboratorial prototype.

The video signal from the camera is transmitted to a personal computer, where the captured image is processed, through a »FireWire« interface [11].

Another part of the hardware equipment used is a laser, which we used to project a laser dot. We used an ordinary commercial laser-pointer.

The software equipment was developed in the Microsoft Windows XP operating system environment and Microsoft .NET Framework, using the C# programming language.

Since the system is currently in the phase of a laboratory prototype, the camera used in performing the tests has been firmly attached to a camera stand, to which we also fixed the laser using an aluminum profile (Fig. 1). In the future we plan to attach the system under identical mounting conditions (identical declination angle of the camera and the laser), to a voice controlled wheelchair, where it will contribute to better safety.

3. PRINCIPLE OF OBSTACLE DETECTION WITH LASER TRIANGULATION

At the beginning we calibrate the system using two fixed points which are at a pre-measured distance from the system itself. The source of laser light has to be fixed above the camera, although the information about its exact position and declination angle is unimportant. The position of the laser source will be treated indirectly through the straight line (of the laser beam), traveling through points T_1 and T_2.

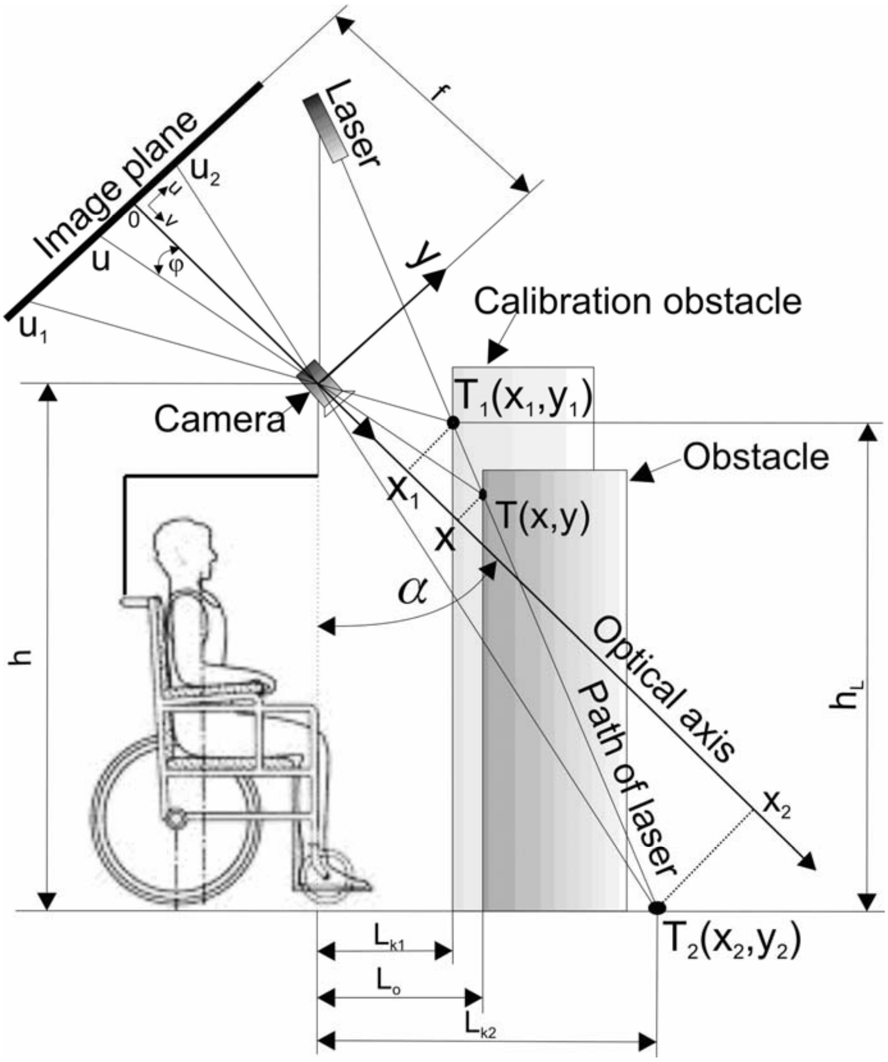

Figure 2: Geometry of the system.

Points T_1 and T_2 are calibration points. We have to ensure that T_1 is situated below the optical axis of the camera, while T_2 is situated above it. The calibration process is performed by first projecting the laser dot to a distance Lk_2 in front of the system and then determining the position of the dot (u_2) in the image captured by the camera. We also have to know the position of T_2`s projection on the optical axis of the camera. By deriving the equations describing geometrical laws (Fig. 2), we obtain equation for calculating the right angle projection of the point T_2 to the optical axis of the camera:

$$x_2 = \cos\alpha \cdot h + \sin\alpha \cdot L_{k2}, \tag{1}$$

where h represents the altitude of the camera above the base, Lk_2 represents the distance between the system and the position of the projected laser dot, while α is the declination angle of the camera.

In the next phase we place the so called calibration obstacle at the distance Lk_1 in front of the system. In this case laser dot is projected on the obstacle in the point T_1. We capture the image with camera and then determine the position of that point (u_1) in the captured image. We also have to determine the projection of the calibration point on the optical axis. Considering the geometry characteristics we get the equation:

$$x_1 = \cos\alpha \cdot (h - h_L) + \sin\alpha \cdot L_{k1}, \tag{2}$$

where Lk_1 represents the distance between the system and the calibration obstacle, while h_L represents the distance between the point where dot is projected on the obstacle and the base.

For the similar triangles we can write:

$$\frac{y_1}{x_1} = \frac{u_1}{f} \qquad and \qquad \frac{y_2}{x_2} = \frac{u_2}{f} \tag{3}$$

where f is the focal length of the camera.

Now we can write down the equation for the straight line traveling through points T_1 and T_2, together with the directional coefficient of this straight line:

$$k = \frac{y_2 - y_1}{x_2 - x_1} \qquad and \qquad y_2 = kx_2 + n \Rightarrow n = y_2 - kx_2 \tag{4}$$

If we consider equation (3) in the equation (4) we get:

$$k = \frac{u_2 x_2 - u_1 x_1}{f(x_2 - x_1)} \qquad and \qquad n = \frac{u_2 x_2}{f} - kx_2 \tag{5}$$

When an obstacle appears in front of the system, the laser dot is projected on that obstacle in the point $T(x,y)$. From the captured image we then obtain the information about the position (u) of this point in the image. Considering the geometry of similar triangles (Fig. 2), we can write for the point $T(x,y)$:

$$\frac{y}{x} = \frac{u}{f} \Rightarrow y = \frac{u}{f} x \tag{6}$$

and for the straight line describing the laser path:

$$y = kx + n \tag{7}$$

If equate equation (6) and equation (7), isolate *x* and then in this new equation consider equations (5) we get:

$$x = \frac{(u_1 - u_2)x_1 x_2}{u(x_2 - x_1) - u_2 x_2 + u_1 x_1}. \tag{8}$$

Now we write down equation (8) in the form:

$$x = \frac{k_1}{uk_2 - k_3}, \tag{9}$$

where k_1, k_2 and k_3 are the calibration constants as follows:

$$\begin{aligned} k_1 &= (u_1 - u_2)x_1 x_2 \\ k_2 &= x_2 - x_1 \\ k_3 &= u_2 x_2 - u_1 x_1. \end{aligned} \tag{10}$$

Equation (9) therefore enables us to calculate the distance between the position of the camera and the obstacle (the distance through the optical axis of the camera). In our application we are interested in the baseline distance between the camera and the obstacle. Considering the equations describing geometrical laws, we get a formula for calculating the distance to the obstacle in the following two cases:

- if the projected point is above the optical axis of the camera:

$$L_o = x(\sin\alpha + \cos\alpha \cdot tg\varphi) \tag{11}$$

- if the projected point is below the optical axis of the camera:

$$L_o = x(\sin\alpha - \cos\alpha \cdot tg\varphi) \tag{12}$$

providing that:

$$tg\varphi = \frac{u}{f}. \tag{13}$$

4. IMAGE PROCESSING

As it has already been mentioned, we need information about the position of the laser dot projection in the image captured by the video camera. In usual circumstances there are also other objects beside the projected laser dot present in the captured image. By using appropriate image processing we can separate the laser dot from the background and determine its position.

In the following passage we will describe image processing algorithm used for determining the position of one projected laser dot in the image. The algorithm has been designed in a manner that enables us to easily expand it and use it for determining the position of obstacles using several laser dots or a laser line. In a case of using a projected laser line, this can be imagined as a large number of laser dots projected close to each other. We can then determine the position of each individual dot by using the discussed algorithm.

Figure 3 represents a block scheme of the image processing principle.

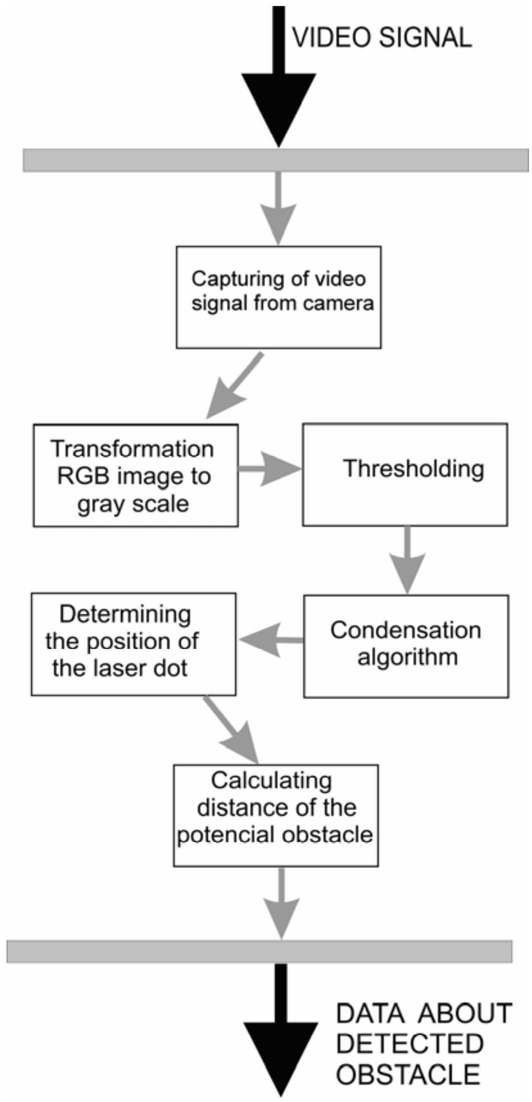

Figure 3: Block scheme of image processing.

Capturing of video signal is performed with the help of dshow [7] software framework. Snapshots of the video are performed in *30 ms* intervals. Each snapshot represents an image which is then digitally processed.

Since in the present paper we will limit ourselves to monochromatic images, we have to transform the obtained colour images by using the transformation:

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{14}$$

In the equation (14) R represents red component, G represents green component and B represents blue component for each individual image element (from now on referred to as pixel), while *Y* represents the calculated grey scale value.

As we proceed we have to separate the object of interest from the background. This is accomplished by using a threshold method [10]. A common problem in this process is that noise can lead to errors. We can improve the results obtained with the threshold method by processing the image with condensation algorithm. Once the information about the position of the projected laser dot has been acquired, we can determine the presence and distance of a potential obstacle.

4.1 Thresholding

As mentioned previously, this method enables us to separate the illuminated object of interest (in our case laser dot) from the background. The method is based on a comparison of the gray scale value of each pixel with a certain threshold value. If the gray scale value of the pixel exceeds the value of the threshold, we set the pixel value to 255 (maximum luminance of a point), while pixels with gray scale value below the threshold value are set to 0 (a dark point). In this way we obtain a binary image with values 255 and 0. The efficiency of this method largely depends on the appropriate selection of the threshold value. We have chosen this value experimentally. Figure 4 represents a flow chart of the image processing employing the threshold method.

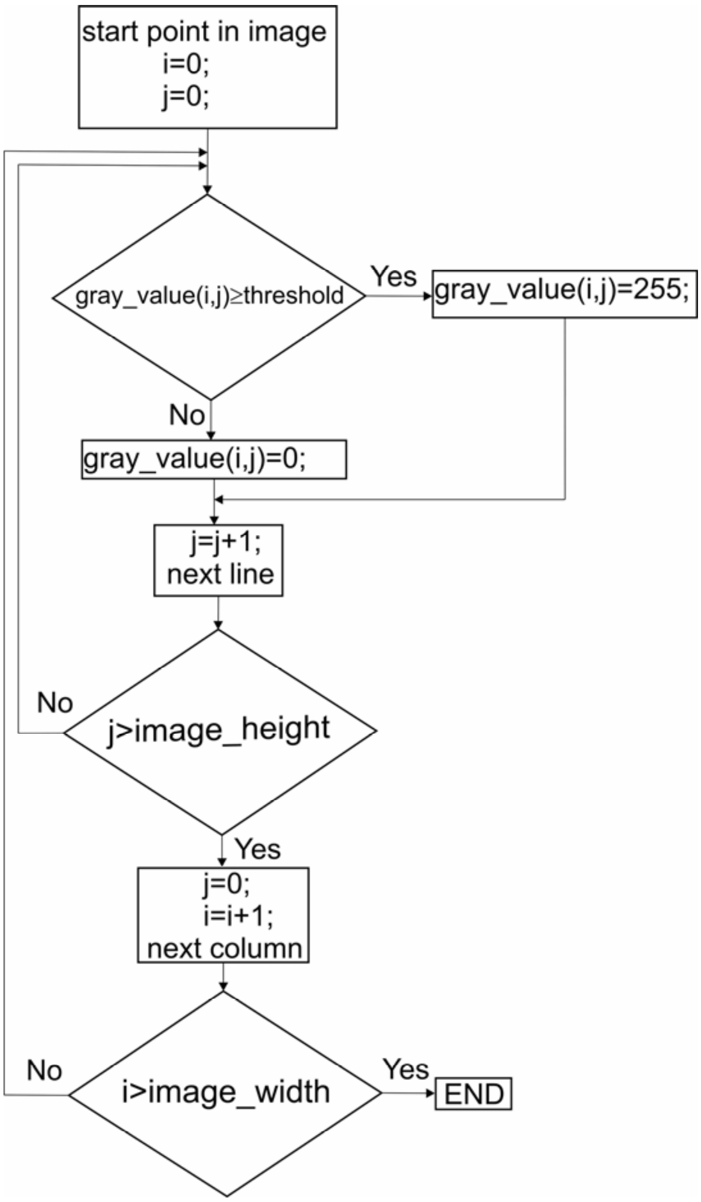

Figure 4: Flow chart of threshold method.

In our case, the laser illuminated sector represents an area of extreme possible gray scale values (values close to 255), therefore a method with a single threshold yields good results. Fig. 5 represents the user interface of the application implemented in C# programming language, with result of separating the object from the background.

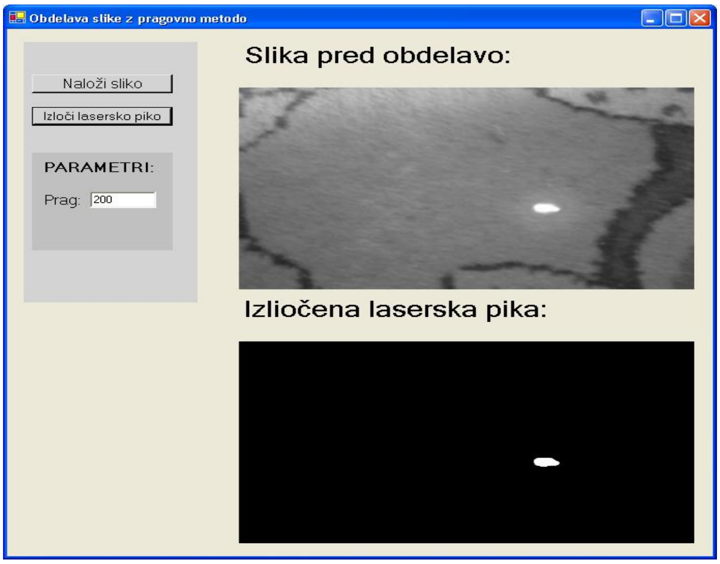

Figure 5: User interface of application for separating the object from background.

4.2 Condensation algorithm

The aim of the threshold method is to separate only the laser dot from the image background. In other words, all the picture components lying inside the ellipse representing the laser dot have to be set to a maximum value 255, while all other components are set to value 0. Conditions that are not ideal, can often lead to errors. By using the condensation algorithm we can improve the results obtained from image processing using the threshold method. The condensation performed by proceeding through the binary image column-wise and with a predetermined step, capturing a certain number of pixels (frame) in each step. Size of the step and the frame is determined experimentally, relatively to the size of the projected dot. Each captured framework is checked for the number of pixels with value 255. If these exceed half of the total number of pixels in a frame, we substitute the entire frame with one pixel with value 255 (Fig. 6). In the opposite case we substitute the entire frame with one pixel with value 0.

Fig. 6 shows a schematic representation of column n from a binary image in which white colour represents maximum gray scale values. We have deliberately added some noise to the column. For a demonstration of the algorithm's functioning we have set the size of the frame to 5 and step to 3.

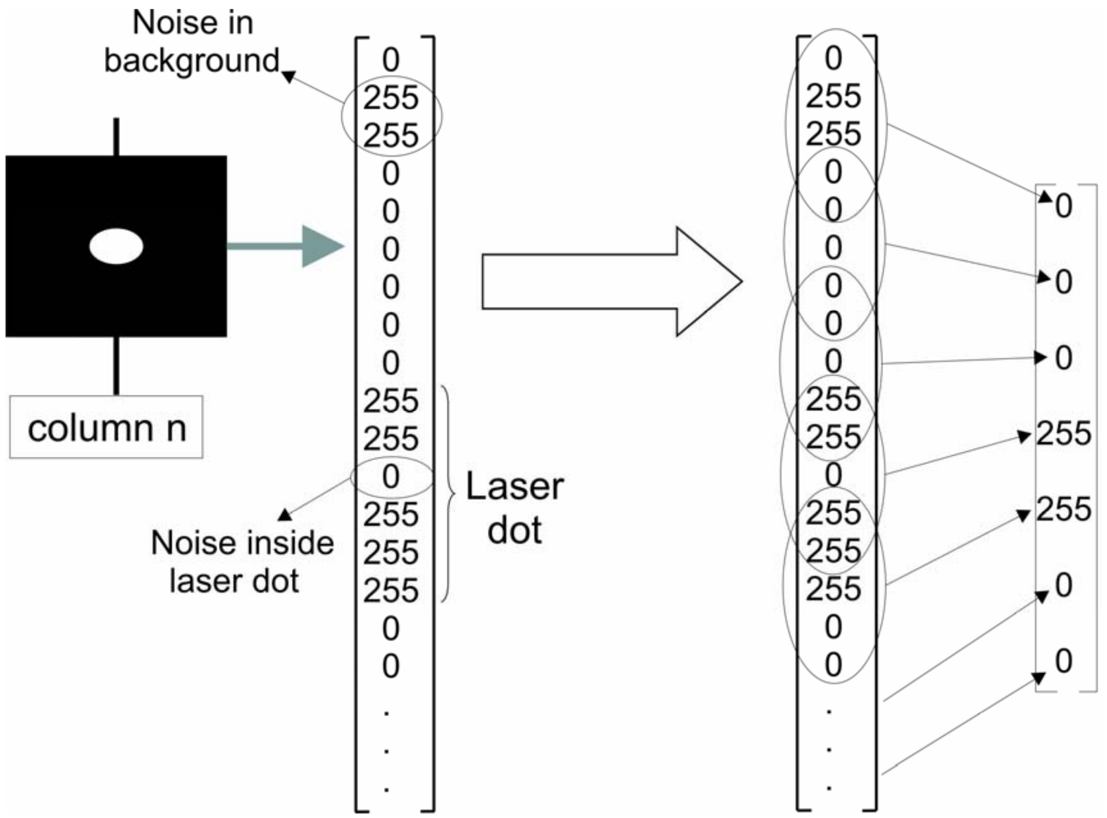

Figure 6: Schematic representation of the condensation algorithm's functioning.

As a result, we have obtained a binary image that was reduced height-wise, and from which the eventual noise was removed. As seen from Figure 6, the laser dot in the column n comprised 6 pixels before the condensation and only 2 pixels after the condensation.

4.3 Condensation algorithm

At the end we examine the binary image for the column with the highest number of pixels with value 255. Since the projected laser dot is elliptical in shape, the column with the highest number of pixels with maximum luminance value represents the center position of the dot in horizontal direction. If we then search for the line with the highest number of pixels with value 255, we can determine the center position of the projected laser dot in vertical direction. Consequently we obtain the information about the position of the center of the projected laser dot.

5. ACQUIRING ADDITIONAL INFORMATION ABOUT OBSTACLES BY USING FUZZY EDGE DETECTION

By using edge detection method we can define optical edges in an image. Edge in an image is defined by characteristic alterations of gray scale or colour component values in a specific direction. Edge detection method therefore detects optical edges which represent obstacles, as well as edges in otherwise leveled surfaces without obstacles, detected by the method because of characteristic alterations in gray scale and colour component values. This means that by using exclusively edge detection method, we can not obtain reliable information about the potential presence of obstacles. We also can not obtain information about the distance of a potential obstacle.

The algorithm for detecting the position of the laser dot in image has been expanded by the method of fuzzy edge detection. First we use the projected laser dot to detect the obstacle and determine its position, then by employing the method of fuzzy edge detection we detect edges in the vicinity of the projected dot, which provides us with additional information about the size and shape of the obstacle.

5.1 Fuzzy edge detection

The algorithm detects and sharpens the edges in an image. Advantages of the fuzzy detector are its low computing cost and low sensitivity to noise. Eight values are calculated for each pixel in the image relatively to its eight neighborhood pixels. Lets look for example at pixel in position p5. For the 3x3 neighborhood of the central pixel in position p5, we calculate the differences between the gray scale value of the pixel p5 and its neighborhood pixels (equation (15)).

$$\begin{array}{ll} X1=Xp1-Xp5 & X2=Xp2-Xp5 \\ X3=Xp3-Xp5 & X4=Xp4-Xp5 \\ X5=Xp6-Xp5 & X6=Xp7-Xp5 \\ X7=Xp8-Xp5 & X8=Xp9-Xp5 \end{array} \qquad (15)$$

p1	p2	p3
p4	p5	P6
p7	p8	p9

X1	X2	X3
X4	0	X5
X6	X7	X8

Figure 7: 3x3 neighborhood of pixel p5 and calculated differences.

In this way, each pixel in the image is presented by an 8-dimensional vector (X1, X2, ... X8). These vectors are the entrance points into our fuzzy system. The entire fuzzy edge detection algorithm is described in detail in [4].

6. RESULTS

The entire application was designed in the Microsoft .NET framework using C# programming language. Prototype of the application for obstacle detection with the help of laser triangulation and the supplementary application of the fuzzy edge detection method were tested on a personal computer with a 1,4 GHz processor and 256 MB of physical memory. The system of obstacle detection with active laser triangulation has been tested by using a projected laser dot. During the procedure, we have processed a new captured image every 30 ms, which is time-wise efficient enough to be realized in an actual application. Analysis of the obstacle detection reliability has pointed to the utter importance of a proper threshold value selection. In our tests we have chosen a high threshold value (220). We have also reduced the contrast of video by using an in-build video camera feature, which resulted in only the most illuminated objects being visible in the captured images. We have successfully tested obstacle detection in different conditions, obtaining good results. The algorithm of processing has also been tested in images to which several other projected laser dots or a projected laser line was digitally added. This yielded similarly good results, but also led to a greater time consumption, which lengthened the intervals in which the obstacle detection is being performed.

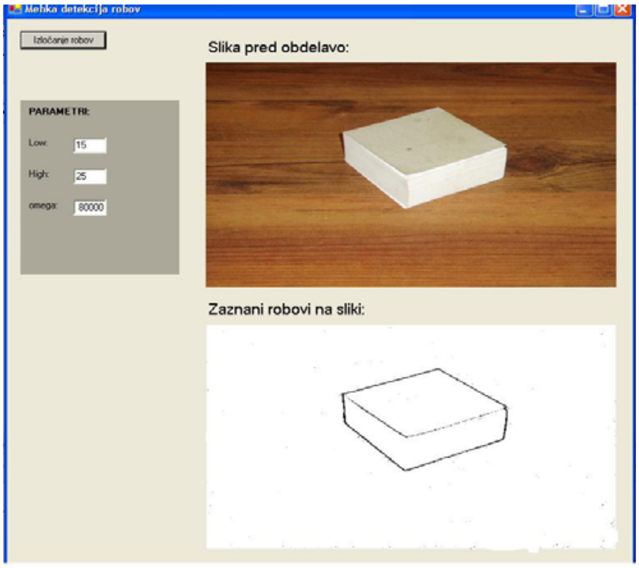

Figure 8: Detected edges represent the edges of an obstacle.

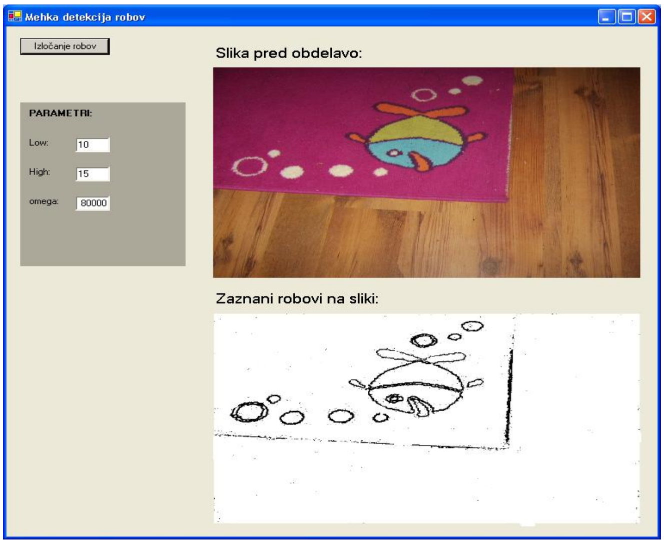

Figure 9: Detected edges represent the edges of an obstacle.

Analysis of the fuzzy edge detection method has shown that the method is very efficient in detecting optical edges in an image even in a case of noise. Figure 8 shows results of the optical edge detection method in a case where optical edges represent the edges of an obstacle, while figure 9 shows the results of edge detection method in a case of edges being detected because of alteration in colour components. The algorithm could be further expanded by evaluating and obtaining useful information (width, height of the obstacle...) from the detected optical edges, but this would consequently lead to even greater time consumption.

7. CONCLUSIONS

The method of obstacle detection with a projected laser dot has proved to be very satisfactory. The prototype presented in the present paper has been developed for the purpose of testing and evaluating the efficiency of the system. In the future it would be reasonable to implement the software of the system to a card with a capable digital signal processor (DSP), which would provide an economically and technologically suitable solution for the usage in a wheelchair. The efficiency of a DSP card will enable us to utilize a larger number of projected laser dots or a projected laser line in real time. This will enable us to perform obstacle detection in the whole area in front of the wheelchair and also enable obstacle detection and analysis with edge detection.

REFERENCES

[1] Tetičkovič, A.; Klančnik, S. (2005). Razpoznava govora z usmerjeno nevronsko mrežo, *Proceedings of the 14th International Electrotechnical and Computer Science Conference* ERK'05, 387-389
[2] Tehnical University of Crete, Computer vision, *www.intelligence.tuc.gr*, accessed on 05-03-2007
[3] TUDelft. Image processing fundamentals, from *www.ph.tn.tudelft.nl/Courses/FIP*, accessed on 05-02-2007
[4] Liang, R. L.; Basallo, E.; Looney, C. G. (2001). Image Edge Detection with Fuzzy Classifier, *Proceedings of the ISCA 14th international Conference Las Vegas*, pp. 279
[5] Figueroa, J. F.;Everett, H. R. (2003). Distance Measuring and Proximity Sensors, Bishof R. H. (Editor), *The Mechatronic Handbook*, CRC Press
[6] Sony. Tehnical specification, from *www.sony.si,* accessed on 08-02-2007
[7] Direct show. Video capture, from *www.videohelp.com*, accessed on 05-03-2007
[8] Woods W. J. (2006). *Multidimensional signal, image, and video processing and coding*, Elsevier Inc., New York
[9] Bajić, I. V.; Woods, J. W. (2003). Domain-Based Multiple Description Coding of Images and Video, *IEEE Trans. Image Process*., 1211-1225
[10] Gonzalez, R. C.; Woods, R. E. (1993). *Digital Image Processing*, Addison-Wesley, Boston
[11] Iglesias, T.; Salmon, A.; Scholtz, J.; Hedegore, R. (2006). Camera Computer Interface, Hornberg, A. (Editor), *Handbook of Machine Vision*, Wiley-VCH, Weinheim, 427-508