Enhanced Algorithm for Obstacle Detection and Avoidance Using a Hybrid of Plane To Plane Homography, Image Segmentation, Corner and Edge Detection Techniques

N.A. Ofodile[1], S. M. Sani[2] *PhD, MSc(Eng), BSc(Hons)Eng*
[1] *(NAF Research and Development Centre, Air Force Institute of Technology, Nigeria)*
[2] *(Department of Electrical and Electronics Engineering, Nigerian Defence Academy, Nigeria)*

Abstract: *This paper presents the implementation as well as simulated results of the enhanced algorithm for obstacle detection and avoidance using a hybrid of plane to plane homography, image segmentation, corner and edge detection techniques. The key advantages of this algorithm over similar ones are:*

(i) elimination of false positives obtained by the image segmentation technique as a result of which obstacle detection becomes more efficient,
(ii) reduction in the presence of unreliable corners and broken edge lines in high resolution images which may result in poor homography computation and image segmentation respectively,
(iii) elimination of lack of depth perception hence the system provides and evaluates depth and obstacle height properly without planar assumptions,
(iv) significant reduction in processing power.

Keywords: *obstacle detection and avoidance, plane to plane homography, image segmentation, corner detection, edge detection.*

I. Introduction

Obstacle avoidance is a fundamental requirement for autonomous mobile robots and vehicles. Due to human error, the obstacles may not be detected on time or the divert signal meant to change a vehicle's direction may be interrupted deliberately by jamming and the vehicle could be destroyed as a result. The aim of this system is to develop an optimized obstacle detection and avoidance system algorithm for use onboard an unmanned ground vehicle. The system makes use of a camera (image acquisition device) placed on board a UGV connected to an onboard processing unit. The processing unit will perform the functions of obstacle detection, avoidance and motor control of the UGV. Specifically, the system will focus on the design and implementation of the obstacle detection and avoidance based on the processed images obtained from the cameras. The sensors of obstacle detection systems are built on different technologies. These technologies are [1]:

i. infrared sensors,
ii. common Radio Detection and Ranging (radar) sensors,
iii. microwave-based radar,
iv. digital cameras,
v. laser detection and ranging (ladar).

Apart from digital cameras and ladar, the other technologies are based on electromagnetic radiations or radio frequency signals which have the following impairments;

i. reduction in signal quality due to climatic and weather conditions,
ii. reduction in signal quality due to scattering nature of electromagnetic signals as they hit certain material surfaces.

RostislavGoroshin[2] developed an **Obstacle detection using a Monocular Camera** focused basically on a single algorithm. The algorithm processes video data captured by a single monocular camera mounted on the UGV. They made the assumption that the UGV moves on a locally planar surface, representing the ground plane. However the monocular camera could not provide and evaluate depth and obstacle height properly due to lack of depth perception which is common with planar assumptions and multicolor images could not be properly segmented since the original algorithm focused on segmentation techniques for less colored (single, dual or tri) images.

SyedurRahman [3] worked on **the development of Obstacle Detection for Mobile Robots Using Computer Vision**. The system used Multi-view relations on epipolar geometry and edge detection to find point correspondences on edges between the images and then uses planar homography to compute the heights along the contours thereby performing obstacle detection. However, several optimizations need to be made to enhance

the reliability of the method. For example, if an obstacle and the ground get segmented together, epipolar geometry and contour height estimates could be used to detect where the ground ends and where the object starts. A horizontal line can be drawn separating the obstacle and the ground marking them with their appropriate heights. Also, images with better resolution resulted or lead to the presence of more unreliable corners and broken edge lines which may make matters worse during the homography computation.
The processes involved in the design of this system include:

i. modeling the Video based Obstacle Detection and Avoidance System using SIMULINK/MATLAB Modeling Software; an easy to use tool used for simulation and parameter optimization
ii. designing the Video Processing and Image Processing Algorithm for the Video stream and apply the designed obstacle detection algorithm to the video stream.
iii. implementing the obstacle detection-avoidance system

II. System Implementation

a. PLANE TO PLANE HOMOGRAPHY

Plane to plane homography can be described as a relationship between two planes, such that any point on one plane corresponds to one point in the other plane.Homography simply means an invertible transformation from a projective space that maps straight lines to straight lines.In figure 1.1a, an image of a scene is shown. It contains two points x_1 and x_2 that will be used to show how homography can be used to obtain other point coordinate values on the same image or another image of the same scene.

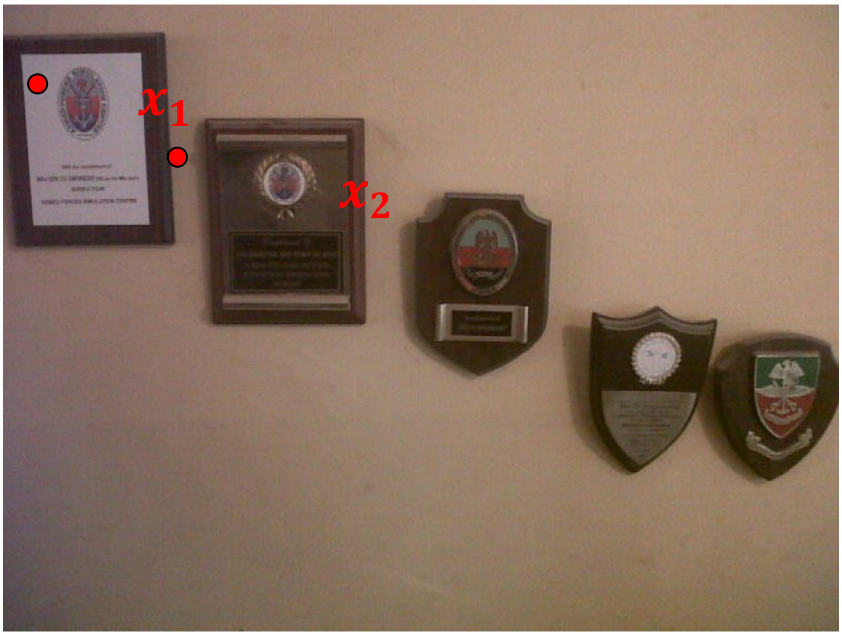

Figure 1.1a: an image of a scene showing x_1 and x_2 image points

In order to get the width of the second plaque from the left, two homography matrices are computed from 2 points with coordinates X_1 and X_2 on the scene and the coordinates x_1 and x_2 on the image. X_1 and X_2 were measured using a metre rule with the lower right corner as reference from the actual scene while $x_1$and x_2 were measured from the image in figure 1.1a. The measured values for X_1, X_2, x_1 and x_2 are given below. All dimensions used in this computation are in centimeters (cm).

$X_1 = [67.5,17.65,1]$
$X_2 = [23.3,21.9,1]$ (1.2)
$x_1 = [13,9.5,1]$ (1.1) (1.3)

$$x_2 = [8.8,5.7,1] \tag{1.4}$$

The homography matrices are computed using the formula MATLAB algorithm containing the formula $X = Hx$

$$X_1[67.5 \quad 17.65 \quad 1] = x_1[13 \quad 9.5 \quad 1]\begin{bmatrix} h1 & h2 & h3 \\ h4 & h5 & h6 \\ h7 & h8 & h9 \end{bmatrix} \tag{1.5}$$

$$X_2[23.3 \quad 21.9 \quad 1] = x_2[8.8 \quad 5.7 \quad 1]\begin{bmatrix} h1 & h2 & h3 \\ h4 & h5 & h6 \\ h7 & h8 & h9 \end{bmatrix} \tag{1.6}$$

Therefore the homography matrices are;

$$H_1 = \begin{bmatrix} 2.5 & 1 & 0 \\ 3.2 & 0.3 & 0 \\ 1.6 & 1.8 & 1 \end{bmatrix} \qquad H_2 = \begin{bmatrix} 1.5 & 1.9 & 0 \\ 1.2 & 0.7 & 0 \\ 3.3 & 1.2 & 1 \end{bmatrix} \tag{1.7}$$

Hence, if the width of the frame on the image in fig 1.1b is 2.5cm and its length is 3cm such that

$$x_3 = [2.5,3,1] \tag{1.8}$$

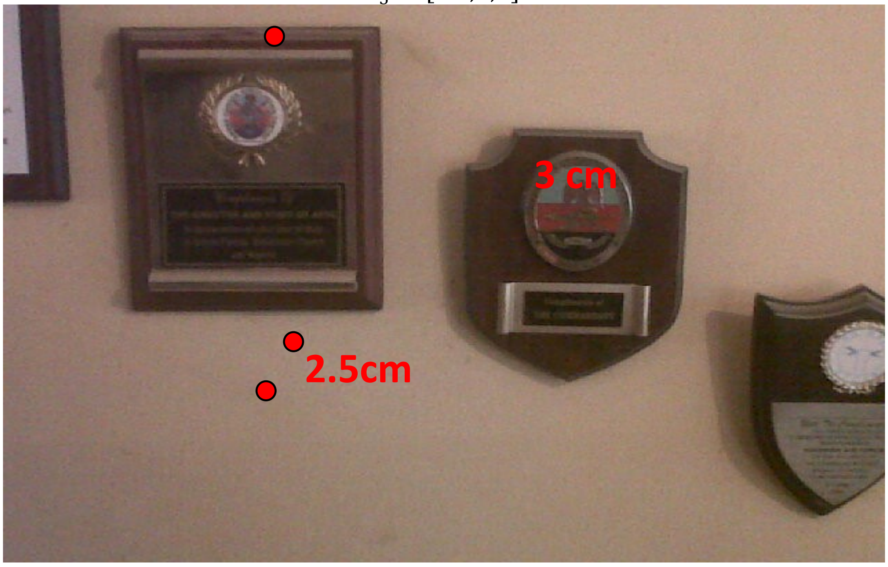

Figure 1.1b: another image of the scene showing x3 image point

Then the estimated width and length of the plaque when computed using the homography matrix H_2 will result in

$$X_3 = x_3[2.5 \quad 3 \quad 1]H_3 \begin{bmatrix} 1.5 & 1.9 & 0 \\ 1.2 & 0.7 & 0 \\ 3.3 & 1.2 & 1 \end{bmatrix} \tag{1.9}$$

$$X_3 = [10.65,8.05,1] \tag{1.10}$$

The actual width and length of the plaque when measured on the actual scene using a metrerule is 10.9cm and 7.77cm respectively. The plane to plane homography completely depends on its structure to determine relevant information but this project will combine the camera's internal parameters and relative pose that will use simple lens formula given as [4]:

$$\frac{1}{focal\ length} = \frac{1}{Object\ height} + \frac{1}{Image\ height} \tag{1.11}$$

to compliment the homography computations where the homography may not be available.

b. Image Segmentation.

Image segmentation can also be called warping. While creating the warped image, the warped coordinates ***x***of each pixel is found using ***X = Hx.*** when given the coordinates ***X***on the first image and the homography matrix H [5]. It is similar to the computation above. However this means that there may be pixels on the warped image which are not warped for any pixels in the first image and there may be pixels that are warped for more than one pixel from the first image. These problems are solved using interpolation. Blank pixels are simply filled up by averaging the intensities of their non-blank neighbours. Pixels that are warped positions for more than one pixel on the first image have the average intensities for all the corresponding pixels from the first image.

Assuming the plane to which the homography corresponds to is the ground, the warped image and second image should be identical except for parts of the scene that are above the ground plane (i.e. obstacles). The difference between intensities of corresponding pixels between the warped image and second image is used to detect objects or obstacles.

c. Canny Edge Detection.

The Canny edge detection algorithm is also known as the optimal edge detector. In the paper written by John Canny titled "A Computational Approach to Edge Detection" [6], he followed a list of criteria to improve the three methods of edge detection listed earlier. The criteria are;

i. the first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be no responses to non-edges;

ii. the second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum;

iii. a third criterion is to have only one response to a single edge. This was implemented because the first two criteria were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the non maximum suppression regions. The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it has made an edge but if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above the second threshold.The canny edge detector is used for this work because of its optimal characteristic and the solution it gives to the three criteria earlier stated.

d. THE HARRIS CORNER DETECTOR.

This was chosen because the Harris corner detector provides variation in intensity of all pixels based on orientation in the image causing a great reduction in the noise response obtained. It is also well suited for a video image analysis application (i.e. real-time) using minimum processing requirement.

Here, the local structure matrix is smoothed by a Gaussian iteration. That is,

$$C_{Harris} = w_G(\sigma) \otimes \begin{bmatrix} \left(\frac{\partial y}{\partial x}\right)^2 & \frac{\partial y}{\partial x}\frac{\partial y}{\partial x} \\ \frac{\partial y}{\partial x}\frac{\partial y}{\partial x} & \left(\frac{\partial y}{\partial x}\right)^2 \end{bmatrix} \qquad (1.12)$$

where$w_G(\sigma)$is an isotropic Gaussian iteration, standard deviation σ, and the operation ⊗ denotes convolution. A measure of the corner response at each pixel coordinates (x,y) is then defined by

$$r(x,y) = \det(C_{Harris}(x,y)) - \kappa\ (\mathrm{trace}(C_{Harris}(x,y))^2, \qquad (1.13)$$

where k is an adjustable constant and C_{Harris} (x,y) is the 2 by 2 local structure matrix at coordinates (x, y). To prevent too many corner features lumping together closely, a non-maximal suppression process on the corner response image is usually carried out to suppress weak corners around the stronger ones. This is then followed by a thresholding process. Altogether, the Harris corner detector requires three additional parameters to be specified: the constant k, the radius, d, of the neighbourhood region for suppressing weak corners, and the threshold value t [7]. The corner response r can be written as

$$r = \lambda^2 \left(\alpha - \kappa\ (1+\alpha)^2\right) \geq 0$$

OR $r = k \leq \frac{\alpha}{(1+\alpha)^2}$ (1.14)

Larger value of k corresponds to a less sensitive detector and yields less corners; smaller value of k corresponds to a more sensitive detector and yields more corners.

The system analysis using the four methods/algorithms mentioned above is done using the sampling method Random Sampling Consensus (RANSAC). The RANSAC (Random Sampling Consensus) algorithm is aniterative technique that generates candidate solutions by using the minimum number observations (data points) required to estimate the underlying model parameters [8]

III. Results And Analysis

This involves testing the algorithm written for the obstacle detection using two images. The test carried out here determined;

i. the accuracy of each obstacle detection method used in the design of the obstacle detection system and the accuracy of the obstacle detection system designed,

ii. the effect of certain factors (object and floor textures) on the obstacle detection system developed.

Two images of the same scene are presented as shown in figure 1.2.

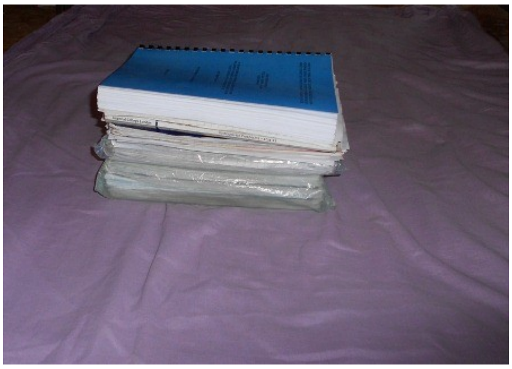

Figure 1.2a Image with stack of papers

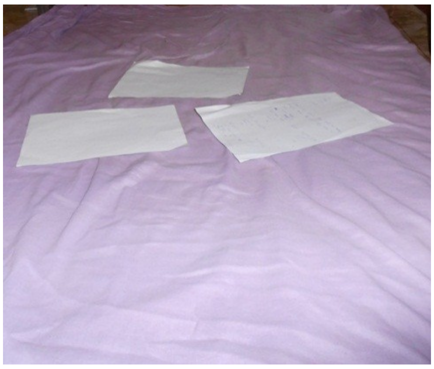

Figure 1.2b image with single paper

One of the images contains a stack of papers which represents an obstacle while the other contains single papers placed on the floor which does not constitute an obstacle. The images were processed using Video Image Processing toolbox of Simulink with each of the obstacle detection methods as well as with the combined obstacle detection system developed.

The results obtained from the effects of the individual obstacle detection methods are outlined below:

i. Corner detection and Edge detection:

The Harris corner detector was used to perform the initial task of locating corners. The results obtained showed that a large number of corners could be found. Also, even when the ground plane and the object were very similar in colour, the Harris detector was able to locate the corners. It was observed that the Canny edge detector detected edges based on the texture of the materials that make up the image. The more detail in the material texture, the more edges that will be obtained. Figure 1.2a shows the full edge detection image of figure 1.2b.

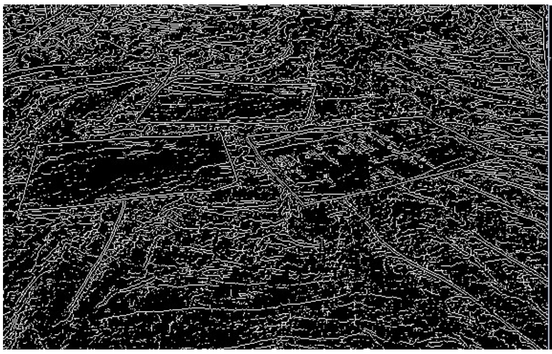

Figure 1.2a Full edge detection

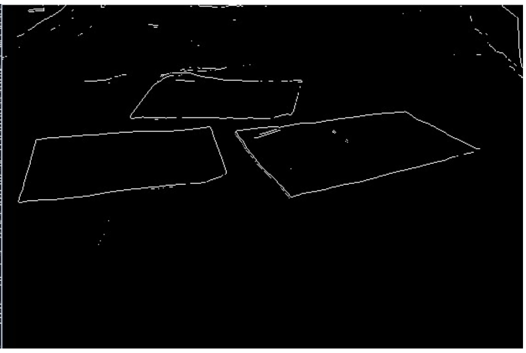

Figure 1.2b Threshold edge detection

The texture of the material representing the ground plane in this image is detailed. The full edge detection image will not be useful for detecting any obstacles on the image. Therefore a threshold scaling factor is introduced such that the original image is compressed to have little detail and the resulting edge detection image as shown in figure 1.2b will leave only edges from objects that are evident on the image.

ii. Plane to Plane homography:

The computation of homography matrix H is entirely dependent on the location of point correspondences between the images. The homography matrix is obtained at the same time as the estimated heights of corners and points along edges. If the computation of the H matrix is accurate, then the computed heights using the algorithm of points relative to the ground on an image would be consistent with their real heights. Figure 1.3a shows selected points on the image in figure 1.2a.

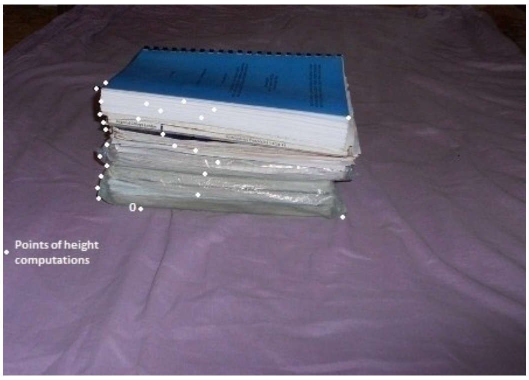

Figure 1.3a Points on the image

Table 1 in the appendix shows the computed heights and their actual heights for 25 different points on the image in figure 1.3a. The real heights were measured with a metre rule while the computed heights were obtained from the computations using the homography matrix

A graph of actual heights and computed heights for the 25 different points on the image of figure 1.3a is plotted and shown in figure 1.3b.

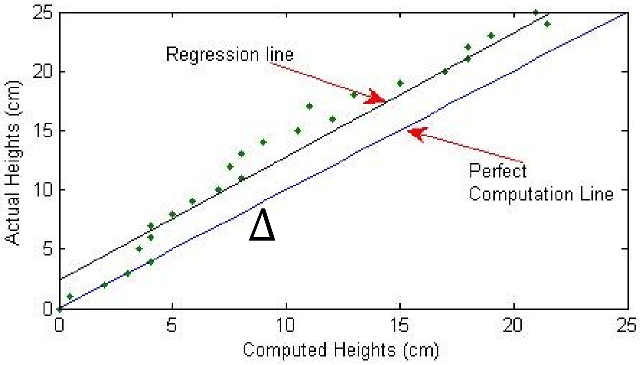

Figure 1.3b Actual heights and Computed heights

The black line in the graph in figure 1.3b is the regression line. A regression is a statistical analysis assessing the association between two variables. It is used to find the relationship between two variables for the purpose of predicting one variable when given the other. This is expressed as an equation [9]:

$$Y = a + bX \qquad (1.1)$$

$$b = (N(\Sigma XY) - (\Sigma X)(\Sigma Y)) / (N(\Sigma X^2) - (\Sigma X)^2) \qquad (1.2)$$

$$a = (\Sigma Y - b(\Sigma X)) / N \qquad (1.3)$$

where

X and Y are the variables.
b = The slope of the regression line
a = The intercept point of the regression line and the y axis.
N = Number of values or elements

There are two broad types of regression models; the linear regression and non linear or robust regression. The Linear regression implements a statistical model where relationships between the independent variables and the dependent variable are almost linear and the Non Linear Regression shows regression as a model of conditional expectation with the conditional distribution of the independent variables given with the more than one dependent variable in the presence of an error term especially when the values used are non-numeric.The linear regression model is used here because the independent variable and dependent variable differences are approximately equal to the average difference which shows that the values are almost linear. Also the values are simply numeric.

Table 2 in the appendix contains the analysis for parameters required to obtain the regression equation.

The Slope "b"

$b = (N(\Sigma XY) - (\Sigma X)(\Sigma Y)) / (N(\Sigma X^2)- (\Sigma X)^2)$
$b = (26*(4317.6) – (247.4)* (325)) / (26 * (3418.06) – (247.4 *247.4))$ (1.4)
$b = 1.151459722$ (1.5)

The Intercept “a”
$a = (\Sigma Y - b(\Sigma X)) / N$
$a = (325 – 1.151459722*(247.4)) / 26$ (1.6)
$a = 1.543417875$ (1.7)

Therefore, the regression equation becomes;
$Y = 1.54 + 1.15X$ (1.8)

The cause of the Δ between the perfect computation line and the regression line represented in equation 1.8 could not be determined in this project. Further works on this project can be done to determine the root cause of the difference obtained.

The regression line (black line) is seen to be quite close to the perfect computation line (blue line). The estimated error or deviation for all the computed heights from the actual heights is about 10% of the actual heights. This deviation and the regression equationis factored back into the main algorithm to recalculate the computed heights and take a tolerance of ±10% during the obstacle avoidance phase (minimum avoidance distance computation). This shows how well the homography of the scene has been computed. Inaccurate point correspondences on the ground plane of an image obtained as a result of too much texture on it, may lead to poor computations of the H matrix. As a result, heights estimated will not be accurate too.

iii. Image segmentation and warping:

The effect of the image segmentation and warping on the two images in figure 1.2 showed that the obstacle (stack of papers) in figure 1.2a was detected. It also detected the singular papers scattered in the image in figure 1.2b as an obstacle. This is however a false positive detection because the papers are on the ground plane and can easily be walked over.

The results obtained from the combined obstacle detection system developed, shows that the false positives detected by the image segmentation and warping is corrected by the algorithm which superimposes the processed image which contains the calculated corner and edge heights onto the homography image obtained from image segmentation and warping and the resultant image is passed through the plane to plane homography phase again so as to eliminate the area of the image with false obstacles leaving only the true obstacles. It also calculates all point correspondences with great accuracy due to the combined effect of the corner, edge detection in the homography matrix computation.

IV. Conclusion

This system used the attendant advantages of corner and edge detection to make up for the shortcomings associated with the homography and image segmentation and warping techniques. The combined system made the obstacle detection system a robust and optimized system.

It is recommended that further work be done on the algorithm of the Program to improve the system detection to perform obstacle detection by reducing estimated error or deviation for all the computed heights from the actual heights. This will remove the tolerance of ±10% during the obstacle avoidance phase (minimum avoidance distance computation).

References

[1]. F Kruse; “Multi Sensor System for Obstacle Detection” Department of Computer Science, Umea University, 2001. http://www8.cs.umu.se/research/ifor/dl/OBSTACLE%20DETECTION-AVOIDANCE/Multi%20Sensor%20System%20for%20Obstacle%20Detection%20in%20Train%20applications.pdfRetrieved 2012-04-16

[2]. RostislavGoroshin; “OBSTACLE DETECTION USING A MONOCULAR CAMERA” School of Electrical and Computer Engineering Georgia Institute Of Technology, 2008 http://smartech.gatech.edu/jspui/1/goroshin_rostislav_200808_mast.pdfRetrieved 2012-03-13.

[3]. SyedurRahman, the development of Obstacle Detection for Mobile Robots Using Computer Vision , 8th European Conference on Speech Communication and Technology, pp. 2197-2200, Geneva, Switzerland, Sept. 1-4, 2003. http://www.wcl.ee.upatras.gr/ai/papers/potamit is1 1.pdf.

[4]. You +1'd this publicly. Undo

[5]. R. Hartley and A. Zisserman: “Multiple View Geometry in Computer Vision”, Cambridge University Press, 2000.SimilarYou +1'd this publicly. Undo

[6]. S. Teller: “Image Segmentation”, Masachussetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory, 1996You +1'd this publicly. Undo

[7]. J. Canny: “A Computational Approach to Edge Detection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986

[8]. C.J. Harris and M. Stephens: “A combined corner and edge detector”, 4th Alvey Vision Conf., Manchester, 1988 SimilarYou +1'd this publicly. Undo

[9]. Z. Chen: “K-means clustering” and “RANSAC normalised 8 point compuatation of F” MATLAB codes, Computer Vision Research, University of York, Department of Computer Science, 2001.

[10]. Ciubotaru-Petrescu, B., Chiciudean, D., Cioarga, R., &Stanescu, D. (2006). Wireless Solutions for Telemetry in Civil Equipment and Infrastructure Monitoring.3rd Romanian-Hungarian Joint Symposium on Applied Computational Intelligence (SACI) May 25-26, 2006. http://www.bmf.hu/conferences/saci2006/Ciubotaru.pdf.

APPENDIX

Table 1: Computed heights and Actual heights measured from points on the figure 1.1a image

Points on the Image	Computed heights (cm)	Actual heights (cm)
Point 1	12	16
Point 2	21	25
Point 3	10.5	15
Point 4	11	17
Point 5	4	6
Point 6	7	10
Point 7	0.5	1
Point 8	9	14
Point 9	7.5	12
Point 10	19	23
Point 11	5.9	9
Point 12	18	22
Point 13	3	3
Point 14	4	7
Point 15	8	13
Point 16	5	8
Point 17	17	20
Point 18	0	0
Point 19	4	4
Point 20	3.5	5
Point 21	24.5	24
Point 22	8	11
Point 23	13	18
Point 24	15	19
Point 25	2	2
Point 26	18	21

Table 2: Analysis for computation of Regression equation

Points on the Image	Computed heights (X)	Actual heights (Y)	X*Y	X^2
Point 1	12	16	192	144
Point 2	21	25	525	441
Point 3	10.5	15	157.5	110.25
Point 4	11	17	187	121
Point 5	4	6	24	16
Point 6	7	10	70	49
Point 7	0.5	1	0.5	0.25
Point 8	9	14	126	81
Point 9	7.5	12	90	56.25
Point 10	19	23	437	361
Point 11	5.9	9	53.1	31.81
Point 12	18	22	396	324
Point 13	3	3	9	9
Point 14	4	7	28	16
Point 15	8	13	104	64
Point 16	5	8	40	25
Point 17	17	20	340	289
Point 18	0	0	0	0
Point 19	4	4	16	16
Point 20	3.5	5	17.5	12.25
Point 21	21.5	24	516	462.25
Point 22	8	11	88	64
Point 23	13	18	234	169
Point 24	15	19	285	225
Point 25	2	2	4	4
Point 26	18	21	378	324
TOTAL N= 26	247.4	325	4317.6	3418.06