



Shadow Removal in Acquired Image for Visual Servoing of Unmanned Ground Vehicles

Kundala Sandeep¹ Eswaran Parthasarathy^{1*} Lakshmi Prathibha²

¹Department of Electronics and Communication Engineering,
 Sri Ramasamy Memorial Institute of Science and Technology, Kancheepuram, India

²Scientist – D', CVRDE (DRDO), Avadi, Chennai, India

* Corresponding author's Email: eswaran.p@ktr.srmuniv.ac.in

Abstract: Vision-guided robotic operation is one of the new concepts in teleoperated Unmanned Ground Vehicle (UGV) for military applications. The objective of the Visual Servoing (VS) is to control the position of the robotic arm using the data such as the distance of an object from the reference frame, or the length and width of the object extracted from the vision sensor. In Image-Based Visual Servoing (IBVS) scheme, position control values are computed from image features directly. This work proposes the technique of shadow removal of the object through the advanced light model from the image, acquired using a monocular camera. The 2D spatial coordinates and Point-of-Contact (PoC) of the object with reference to the ground plane is computed using straight-line equations. PoC analysis is made through mapping of pixel distances to spatial distances to analyses mean pixel distance, and the percentage of error obtained between actual distance and calculated distance computed. Other parameters like the width of the object and, the distance between the camera and object is also estimated. The computed result yields a maximum error of 3 cm and shows 2.6% error for camera fixed at the height of 160 cm with a tilt angle of 50°, which covers an area of 150 cm x 120 cm.

Keywords: Image-based visual servoing, Shadow removal, Single camera, 2D position estimation.

1. Introduction

Visual servoing control for robot manipulator has been adopted in recent days due to its effectiveness to perform complex tasks in the hazardous environment. Visual servoing utilizes the information provided by visual sensors to control robots manipulators.

Visual Servoing can be categorized according to the type of control signal given to the system and camera structure configuration. In control signal configuration, visual servoing for manipulator can be classified as position-based visual servoing (PBVS) [1] and image-based visual servoing (IBVS) [2, 3]. PBVS uses the 3D information of the object such as position and orientation on camera frame, and it is also referred as 3D visual servoing [4, 5]. The control law and error signal are designed using Cartesian coordinates to reduce the position error

and are mapped to manipulator's command. This kind of servoing is very sensitive to calibration or modeling errors, and error minimizing is challenging. IBVS uses the 2D information of the obtained image by a direct projection of the image in image coordinate. In IBVS, signal and control law are directly constructed in the image plane. Hence the errors between observed and desired image feature coordinates [6, 7] are minimized.

In-camera structure configuration, the visual servoing can be classified as eye-in-hand (EIH) configuration [2] and eye-to-hand (ETH) [3] configuration. In an eye-in-hand configuration, the camera is firmly fixed to the manipulator arm's end effector. In an eye-to-hand configuration, the camera is fixed on a stationary place to observe the workspace and the motion of manipulator.

The number of cameras and their configuration used in a visual servoing system remain interesting

research areas which were discussed in [7, 8]. There are three approaches for image-based distance computation namely, stereovision, monovision, and time-of-flight. Stereo vision uses two similar active cameras [9] separated by a distance in either a stand-alone fashion or eye-in-hand configuration for 3D visual servoing to find the disparity map and depth. This method [7, 10] gives high accuracy, but implementing this technique is expensive (as it requires two cameras) and requires high computation time due to simultaneous processing of many images. A single camera is normally deployed in 2D visual servoing lacks in of the depth of information, [11, 12]. However, spatial servoing using a single camera by 3D reconstruction of 2D images has also been studied in the past, [11, 12] which uses eye-in-hand configuration. Depth using single camera [14] is challenging, since it requires taking the global structure of the image into account and also requires prior information about the scene and hence 3D-spatial coordinates cannot be found if the scene is unknown. The time-of-flight depth estimation technique is used to find the depth information by measuring the total time required for light to transmit and reflect from an object. This technique can be a stand-alone or eye-in-hand configuration depending upon the application. Stand-alone configuration is used if a single object to be detected is placed parallel to the camera lens to get the distance between the host and the object. Eye-in-hand configuration is used to get depth information of multiple objects where the light source is required as the end effector of the manipulator's arm has to move to the 2D positions of the objects to get the depth information.

The main feature of IBVS is to identify the object. In extreme environment, object detection may not be accurate due to the presence of shadows. Shadows are one of the major factors degrading the performance of computation. Various methods were proposed to detect and remove shadows in an image. The importance of shadows depends on the type of application. Shadows can also convey useful information about size and shape of objects but create a problem in feature and object detection in an image. Hence, shadow detection and removal is an important pre-processing task for computer vision applications. According to the input video, [15] proposed a two-dimensional vehicle/shadow joint model to estimate its parameter and category, through which the shadow is extracted from the vehicle. However, this kind of method needs so much prior information about the scene which can only be applied to special scenes. Shadows were detected by K-means clustering method on colour

distribution [16] where a darker cluster is classified as shadow region, and the lighter cluster is classified as the non-shadow region. In [17] the shadow region was detected based on the shadow intensity. The shadow is removed by modifying the brightness and colour. Then a filter is used to correct boundaries between sunshine and shadow regions. This algorithm provides good results, but it has constraints on the image surfaces. Most of the works in shadow detection is based on the Illumination-invariant image proposed in [18].

A method to identify and remove shadows from an RGB image is proposed in this work. The illumination-invariant image is used with the original image to locate the shadow edges which are then set to 0. This edge representation is reintegrated followed by lightness recovery which yields a shadow-free image. The innovative work based on graph cut to solve the labelling of shadow and non-shadow regions [19]. Shadow removal is then performed by an image matting approach, and then the shadow-free image is recovered by relighting each pixel in the shadow.

The other important task in IBVS is to convert the image coordinates (pixels) to 2D-spatial coordinates (cm) and distance estimation between the object and manipulator arm using a single camera. Based on the camera properties and geometry applied to the input images, the relation between image coordinates and spatial coordinates is estimated. The in-path and oblique distances of the objects measurement using a single forward-looking camera approach are discussed in [21]. Distance measurement using laser pointer is discussed in [22, 23] which can only be used to get the distance between the camera and object which lies on the principal axis of the camera. An approach for distance measurement from camera to known object where the object is head-on in the line of the optical axis of the camera using the area of pixels covered by the object is presented in [24]. This approach is only applicable if the object's dimension is known and cannot be applied in an unknown environment.

The main advantage of the proposed method is that it can identify any objects in an external environment and determine their sizes and positions accurately compared to other methods even in the presence of shadows.

The paper is organized as follows. Section 2 provides the methodology. Section 3 presents computed results and discussion followed by conclusion in Section 4.

2. Methodology

In the proposed model the IBVS control for robot manipulator under uncalibrated eye-to-hand configuration is used, where the camera is kept at the height of 160 cm and tilted towards the ground at an angle of 50°. Computer Vision and Image Processing Toolbox in MATLAB2016a is used for computational analysis. The manipulator model proposed for eye-to-hand configuration is shown in Fig. 1. The process flow of the proposed method is shown in Fig. 2.

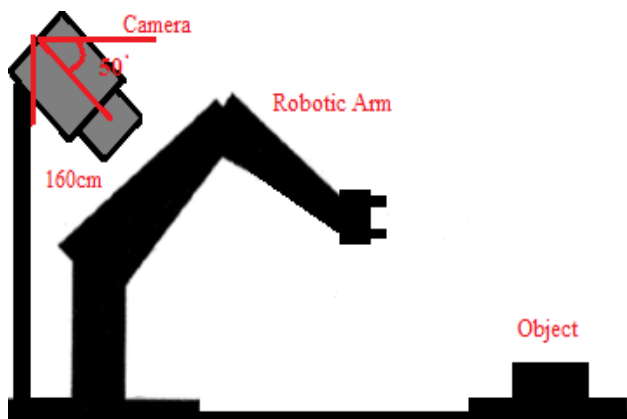


Figure.1 Model of manipulator in eye-to-hand configuration

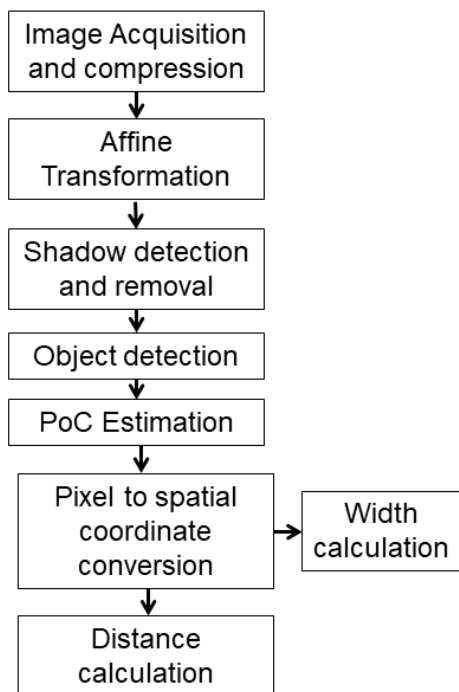


Figure.2 Shadow removal and analysis process flow

2.1 Image acquisition, compression, and affine transformation

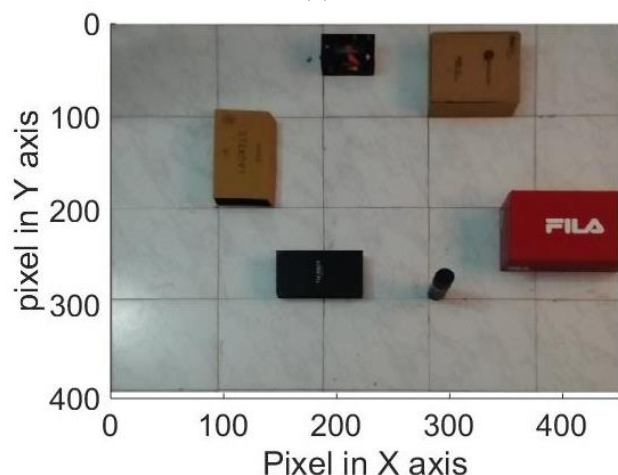
Image acquisition is the first step of any vision system. The acquired image is compressed by a factor of 0.25 for faster processing of the image. Affine transformation and cropping are applied on the compressed image to an exact 150 cm x 120 cm patch as shown in Fig. 3 (a)-(c).



(a)



(b)



(c)

Figure.3 Image acquisition and transformation: (a) Input image, (b) Affine transformed image and (c) Cropped image

The affine transformation is a linear mapping method that preserves points, straight lines, and planes. Sets of parallel lines remain parallel after an affine transformation. The affine transformation technique is suitable to correct the geometric distortions or deformations that occur with non-ideal camera angles. After affine transformation, the region of interest, i.e., the patch which can be covered by the robotic arm is cropped out.

2.2 Shadow removal

In an image, shadows are detected as part of the objects. There are many shadow detection and removal algorithms. In this work advance-light model specified in [25] is adopted and its algorithm is as follows:

The shadow is detected in the image and the 2D mask 'M' is with value 1 is the light region and 0 the shadow region. This mask, later convolved with structure element STREL to obtain the smoothed mask (SM) given in Eq. (1), where

$$STREL = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad (1)$$

The input RGB image is separated into 3 different channels, as R, G and B channels of the image.



Figure.4 Shadow removal: (a) Input images with shadows and (b) Shadow removed images

Obtain averages of both shadow region and light region in R, G, and B channels separately. Calculate the ratio between the average values outside and inside the shadow region.

Multiply each channel of the input image with the ratio obtained in the previous step and then divide the resultant with the product of complementary of SM and respective ratios.

Fig. 4(a) is the input images with shadows. And Fig. 4(b) is the output of the above-mentioned shadow removal algorithm.

2.3 Pixel to spatial conversion

The origin P(1,1) in image coordinate system is considered as S(0,0) in the spatial coordinate system as shown in Fig. 3(c). Each pixel is manually mapped to one to one of their respective spatial distances (in cm), and an equation is formulated for both x and y pixels using the equation of the straight line using 2 points given in Eq. (2) to Eq. (4).

$$S' = \min(\max(S_x), \max(S_y)) \quad (2)$$

$$P' = \text{pixel}(S') \quad (3)$$

$$S(i) = \frac{S' \times (P(i) - 1)}{(P' - 1)} \quad (4)$$

Where S_x denotes the spatial coordinate distance in x-axis, S_y denotes the spatial coordinate distance in y-axis, S' denotes the minimum of the maximum of both S_x and S_y , P' denotes the pixel coordinate value S' (either x or y if S_x or S_y is taken into consideration), $S(i)$ is the spatial coordinate distance value for pixel $P(i)$.

2.4 Distance calculation

The camera is located at (75, 120, 160) cm in actual spatial coordinates. The distance in between the camera and the objects is calculated using two concepts. One, by computing distance between 2 points and the other using Pythagoras theorem. The mean distance between two points is used to find the distance between the object's PoC with the ground to the camera's x-y location $d(i)$. After finding the $d(i)$, the distance between the camera and object is computed using Pythagoras theorem where the known camera height is given as 'H'. The equations are given as follows:

$$d(i)^2 = (x_c - x_o(i))^2 + (y_c - y_o(i))^2 \quad (5)$$

$$D(i) = \sqrt{H^2 + d(i)^2} \quad (6)$$

Where x_c and y_c are x and y coordinates of the camera respectively, $x_o(i)$ and $y_o(i)$ are x and y coordinates of the i^{th} object respectively.

2.5 Width calculation

Width calculation is required to know the size of the object so that the robotic arm can change its claw depending upon the width of the object. Since the objects are of different sizes in an unknown environment and the same claw can't be used for all the sizes. The width of the object is stored in the 'Bounding Box' of 'regionprops' function in pixels. Then Pixel-to-spatial conversion is applied to get the results in spatial distance metrics.

3. Results and discussions

Width calculation is required to know the size of the object so that the robotic arm can change its claw depending upon the width of the object. Since the objects are of different sizes in an unknown environment and the same claw can't be used for all the sizes. The width of the object is stored in the 'Bounding Box' of 'regionprops' function in pixels. Then Pixel-to-spatial conversion is applied to get the results in spatial distance metrics.

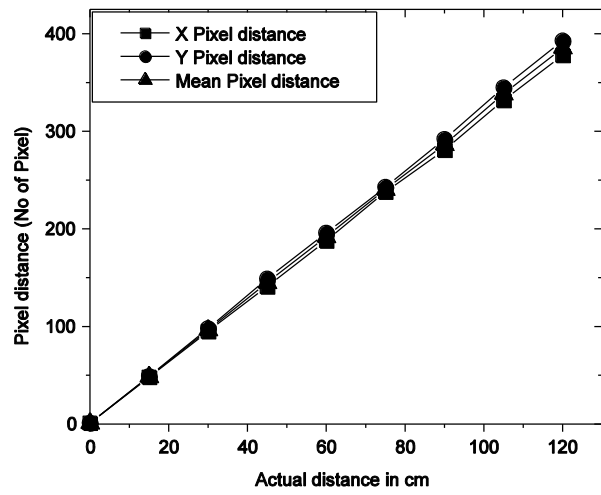
3.1 Mapping of pixel distances to spatial distances

Table 1 provides the one-to-one mapping of pixel coordinates, i.e., pixels in x and y-direction to the actual distance. Mean pixel value is calculated to have a single formula for conversion of both x and y pixel distances to spatial coordinates.

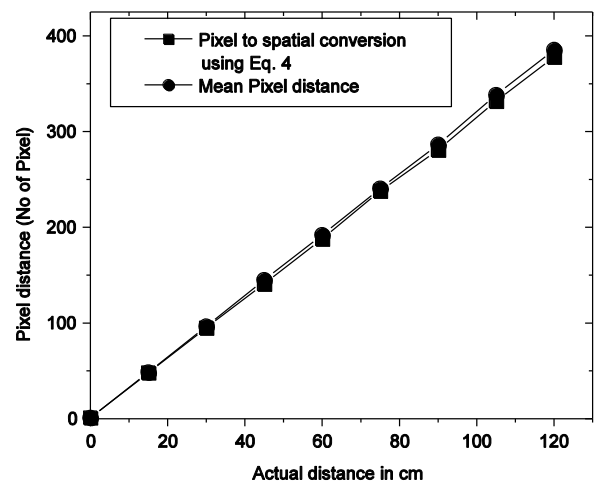
Fig. 5(a) is the plot of x-pixel distances, y-pixel distances and mean pixel distances in pixels against actual distances in cm and Fig. 5(b) shows the deviation between the mean pixel values and the equation used for conversion of pixel coordinates to spatial coordinates. We infer that both the plots are almost overlapping.

3.2 PoC analysis

The comparison of the calculated distance values and actual distance values of the PoC of the objects in x and y-direction from the origin is shown in Fig. 6 and Fig. 7, and the percentage of error obtained is tabulated in Table 2, and the computation value ranges between -3% to +2% and the maximum error measures 3cm approximately.



(a)



(b)

Figure.5 Pixel to spatial conversion: (a) comparison of pixel distances to actual spatial distances and (b) comparison of mean pixel distances with formula used in the project for pixel-to-spatial conversion

Table 1. Comparison between pixel values and actual distances

Actual distance (cm)	X pixel distance (pixel)	Y pixel distance (pixel)	Mean pixel distance (pixel)
0	1	1	1
15	48	49	48.5
30	95	98	96.5
45	141	149	145
60	188	196	192
75	238	243	240.5
90	281	292	286.5
105	332	345	338.5
120	378	393	385.5

3.3 Distance between the camera and objects

Table 3 and Fig. 8 provide the comparison between calculated distances and measured distances of the distance between camera and object. It is inferred that the error % is between -1% to +2% with maximum error measures as 3 cm.

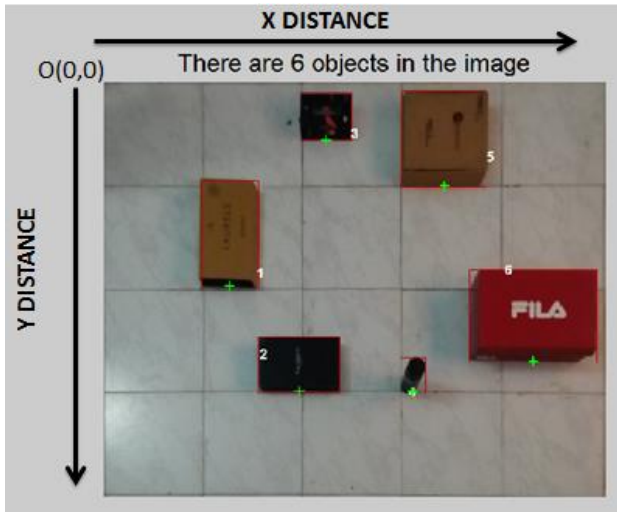


Figure.6 Output image with marked PoC

Table 2. Comparison of calculated and actual 2D positions of the PoC of the objects with ground

Object	X calcul ated (cm)	Y calcul ated (cm)	X actual (cm)	Y actual (cm)	X error (%)	Y error (%)
1	38	59.5	36.8	60.2	-3.26	1.16
2	59	90	57.4	91.1	-2.79	1.21
3	67	16.5	65.2	16.8	-2.76	1.79
4	101.8	30	100.2	30.5	-1.59	1.64
5	92	90.5	90.8	91.4	-1.32	0.98
6	126.5	80.5	126.4	82.4	-0.08	2.31

Table 3. Comparison of calculated and actual distances between the camera and objects

Distance between camera and object	Calculated distance (cm)	Actual distance (cm)	Error (%)
1	175	178	1.69
2	163.5	162	-0.93
3	190.6	191	0.21
4	185	188	1.59
5	163.3	163	-0.18
6	172.2	175	1.6

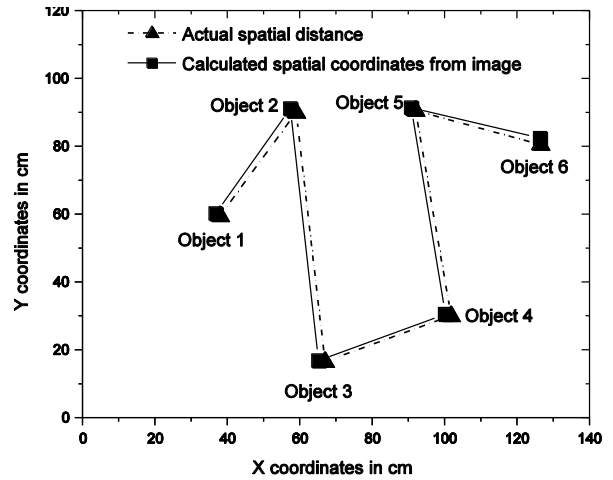


Figure.7 Comparison of calculated and actual 2D coordinates of PoC of the objects in the image

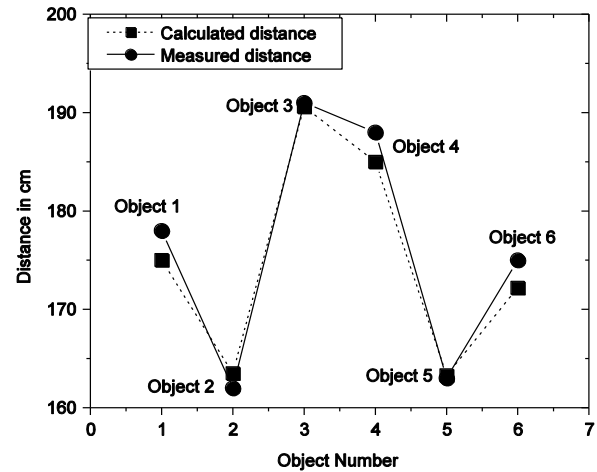


Figure.8 Comparison of calculated and measured distances between camera and object

3.4 Width of the object

Table 4 and Fig. 9 provide the comparison of the calculated and actual width of the object. Depending upon the size of the object the manipulator arm can select the end effector claw on its own to pick up the object. From this analysis, it is inferred that the error in less than 2 cm.

Table 4. Comparison of calculated and measured width of the objects

Object	Width calculated	Width measured	Error (cm)
1	17.165	18	0.835
2	23.407	24	0.593
3	14.98	15	0.02
4	25.28	26	0.72
5	6.866	5	-1.866
6	37.139	36	-1.139

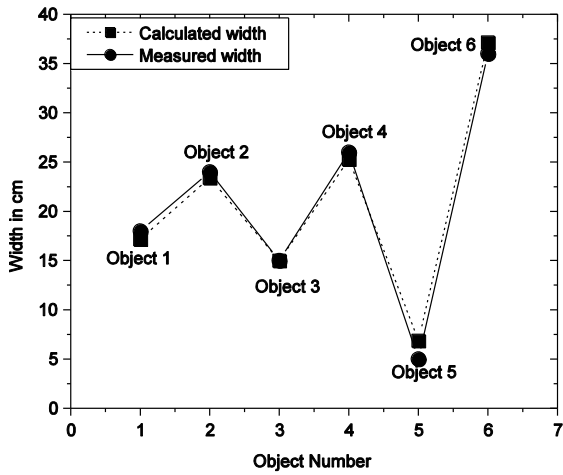


Figure.9 Comparison of calculated and measured width of the objects

3.5 Comparison with latest methods

We compare the proposed method with recently proposed methods in this field. In [26], shadows are used to find the depth-map of the objects in a dark internal environment using a camera and few light sources and then the objects are matched with CAD models to obtain the size and orientation of the objects. Though this method computes the 3D pose of the objects, this method cannot be used in an unknown external environment. In [27], edge detection and pattern matching algorithms are used to detect an object. This method can't be used in an unknown external environment. Though the camera is kept close to the object in [27], the minimum error was 1.4 cm (approx.) in the computation of the size of the object whereas, in the proposed method, the camera was kept at the height of 160 cm and resulted in a maximum of 1.8 cm error. Also in both these methods, the camera is placed parallel to the object and require database containing the object's test images. The proposed distance measurement technique is compared with the method [21] which yields better results than the existing method.

Table 5. Comparison of distance measured using proposed method with Joglekar et al. [21] method

Real distance (cm)	Measured distance (cm) using Joglekar et al. [21]	Measured distance (cm) using Proposed method	Distance error (cm) using Joglekar et al. [21]	Distance error (cm) using Proposed method
30	33.4	31.2	3.4	1.2
60	62.1	59.8	2.1	0.2
90	94.5	90.6	4.5	0.6
120	122.7	118.9	2.7	1.1

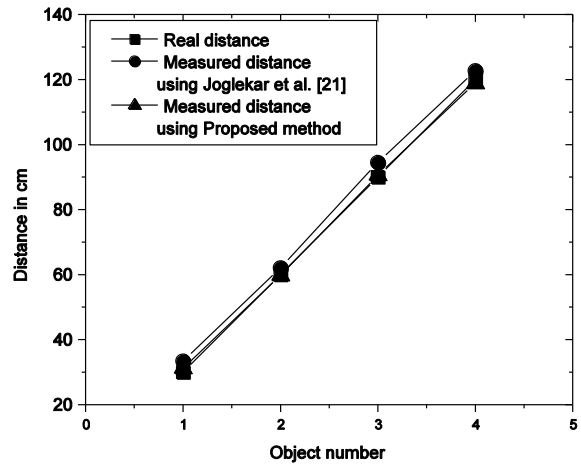


Figure.10 Comparison of proposed method with real distance and Joglekar et al. [21] distance measurement method.

4. Conclusion

In this paper, shadow removal and pixel-to-spatial conversion algorithms are implemented for single camera kept at the height of 160 cm and tilted angle of 50° to cover an area of 150 cm x 120 cm. Shadow removal in IBVS is executed to increase the accuracy of object detection in an external environment, and the pixel coordinates of the object's PoC are converted to spatial coordinates with a maximum error of 2.3% and 3cm, which is within the acceptable error range. These PoC values are given to the system, which is used to control the movement of manipulator's arm in UGV to pick up the object along the sideways at specified points. The future enhancement of the work is to incorporate the feature of 3D position estimation of the object in an external environment.

Acknowledgments

The project was carried out in Combat Vehicles Research and Development Establishment (CVRDE) - DRDO, Avadi.

References

- [1] D. H. Park, J. H. Kwon, and I. J. Ha, "Novel position-based visual servoing approach to robust global stability under field-of-view constraint", *IEEE Transaction on Industrial Electronics*, Vol.59, No.12, pp.4735-4752, 2012.
- [2] H. Wang, Y. H. Liu, and D. Zhou, "Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera", *IEEE Transaction on Robotics*, Vol.24, No.4, pp.843-857, 2008.

- [3] H. Wang, "Adaptive visual tracking for robotic systems without image-space velocity measurement", *Automatica*, Vol.55, pp.294-301, 2015.
- [4] S. Hutchinson, G. D. Hager, and P. Corke, "A tutorial on visual servo control", *IEEE Transaction on Robotics and Automation*, Vol.12, No.5, pp.651-670, 1996.
- [5] L. Weiss, A. Sanderson, and C. Neuman, "Dynamic visual servo control of robots: An adaptive image-based approach", In: *Proc. of IEEE International Conf. Robot. Automation*, Vol.2, St. Louis, pp.662-668, 1985.
- [6] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics", *IEEE Transaction on Robotics Automation*, Vol.8, No.3, pp.313-326, 1992.
- [7] J. Stavnitzky and D. Capson, "Multiple Camera Model-Based 3-D Visual Servo", *IEEE Transaction on Robotics Automation*, Vol.16, No.6, pp.732-739, 2000.
- [8] G. Chesi and K. Hashimoto, "Static-eye against hand-eye visual servoing", In: *Proc. of IEEE International Conf. on Decision and Control*, Vol.3, pp.2854-2859, 2002.
- [9] S. Kumar, D. Gupta and S. Yadav, "Sensor Fusion of Laser and Stereo Vision Camera for Depth Estimation and Obstacle Avoidance", *International Journal of Computer Applications*, Vol.1, No.26, pp.22-27, 2010.
- [10] J. Qian and J. Su, "Online estimation of image jacobian matrix by Kalman-bucy filter for uncalibrated stereo vision feedback", In: *Proc. of IEEE International Conf. on Robotics and Automation*, Washington, D.C., pp.1-5, 2002.
- [11] Y. Shen, Y.H. Liu, and N. Xi, "Adaptive motion control of manipulators with uncalibrated visual feedback", In: *Proc. of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, Vol.1, pp.317-322, Lausanne, Switzerland, pp.1-5, 2002.
- [12] F. Reyes and R. Kelly, "Experimental evaluation of fixed-camera direct visual controllers on a direct-drive robot", In: *Proc. of IEEE International Conf. on Robotics and Automation*, Leuven, Belgium, pp.1-5, 1998.
- [13] C. J. Fang and S.K. Lin, "A performance criterion for the depth estimation of a robot visual control system", In *Proc. of IEEE International Conf. Robotics and Automation*, Vol.2, pp.1201-1206, 2001.
- [14] F. Conticelli and B. Allotta, "Discrete-time robot visual servoing feedback in 3-D positioning tasks with depth adaptation", *IEEE Transaction on Mechatronics*, Vol.6, No.3, pp.356-363, 2001.
- [15] A. Rahman, A. Salam, M. Islam, and P. Sarker, "An Image-Based Approach to Compute Object Distance", *International Journal of Computational Intelligence Systems*, Vol.1, No.4, pp.304-312, 2008.
- [16] A. Yoneyama, C. H. Yeh, and C. C. J. Kuo, "Moving cast shadow elimination for robust vehicle extraction based on 2D joint vehicle/shadow models", In: *Proc. of IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS'03)*, Miami, FL, USA, pp.229-236, 2003.
- [17] M. Baba and N. Asada, "Shadow Removal from a Real Picture", In: *Proc. of SIGGRAPH '03 Conf. ACM SIGGRAPH 2003 Sketches and Application*, San Diego, California, pp.1-1, 2003.
- [18] M. Baba, M. Mukunoki and N. Asada, "Shadow Removal from a Real Image Based on Shadow Density", In: *Proc. of International Conf. on Computer Graphics and Interactive Techniques*, ACM, New York NY, pp.60-60, 2004.
- [19] G. D. Finlayson, S. D. Hordley, and M. S. Drew, "Removing shadows from images", In: *Proc. of 7th European Conference on Computer Vision - Part IV*, pp.823-836, 2002.
- [20] R. Guo, Q. Dai and D. Hoiem, "Single-Image Shadow Detection and Removal using Paired Regions", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, pp.2033-2040, 2011.
- [21] A. Joglekar, D. Joshi, R. Khemani, S. Nair, and S. Sahare, "Depth Estimation using Monocular Camera", *International Journal of Computer Science and Information Technologies*, Vol.2, No.4, pp.1758-1763, 2011.
- [22] K. Muljiowidodo, M. A. Rasyid, N. Sapto Adi, and A. Budiyo, "Vision-Based Distance measurement system using single laser pointer design for underwater vehicle", *Indian Journal of Marine Sciences*, Vol.38, No.3, pp.324-33, 2009.
- [23] N. Shukla and A. Trivedi, "Image-Based Distance Measurement Technique for Robot Vision using New LBPA approach", *International Journal of Novel Research in Electrical and Mechanical Engineering*, Vol.2, No.1, pp.22-35, 2015.
- [24] R. Deepu, S. Murali and V. Raju, "A Mathematical model for the determination of the distance of an object in a 2D image", In:

Proc. of WORLDCOMP 2013, Las Vegas, Nevada, USA, pp.1-5, 2013.

- [25] C. Blajovici, P. J. Kiss, Z. Bonus and L. Varga, "Shadow detection and removal from a single image", *SSIP 2011*, Szeged, Hungary, PP.431-446, 2011.
- [26] A. Agrawal, Y. Sun, J. Barnwell and R. Raskar, "Vision Guided Robot system for picking objects by casting Shadows", *International Journal of Robotics Research*, Vol.29, No.2, pp.1-28, 2010.
- [27] U. D. Kumar, "Vision Assisted Pick and Place Robotic Arm", *Advances in Vision Computing: An International Journal (AVC)*, Vol.2, No.3, pp.9-18, 2015.