Computer Vision Approach for Controlling Educational Robotic Arm based on Object Properties.

Neerparaj Rai[1], Bijay Rai[1], Pooja Rai

[1]Asst.Professor, Electrical & Electronics Dpt,SMIT, Sikkim, India
Email:neerparaj_rai@yahoo.co.in

Abstract: This research presents an autonomous robotic frame work for academic, vocational and training purpose. The platform is centred on a 6 Degree Of Freedom (DOF) serial robotic arm. Two on-board cameras develop a computer vision system for detection and autonomous object/target manipulation placed randomly on a target surface and controls an educational robotic arm to pick it up and move it to a predefined destination. The computer vision is initially used to identify the colour and shape of the object. The system applies Centre-Of-Mass based computation, filtering and colour segmentation algorithm to locate the target and the position of the robotic arm. The proposed platform finds its potential to teach technical courses (like Robotics, Control, Electronics, Image-processing and Computer vision) and to implement and validate advanced algorithms for object manipulation and grasping, trajectory generation, path planning, etc. Experimental results demonstrated the effectiveness and robustness of the system.

Keywords: Computer-vision, robotic arm, autonomous system.

I. INTRODUCTION

Increasing applications and developments in the field of robotics demand trained graduates who are proficient in related technologies. Here arises the need to develop state-of-the-art platforms to serve the purpose. Scientific literature reports many platforms developed specifically for teaching robotic hands [1-3] to help engineering students to stimulate their concepts and to enhance their innovative skills. However such systems do not consider objects of different height for positioning the robotic gripper at the centre of the target. The proposed active-vision system was also tested for objects of varying dimension. This technique uses feedback information extracted from a vision sensor to control the motion of a robot, usually for correcting the gripper position and orientation via the closed-loop control schema [2-5]. R.Szabo presents an application to sort coloured objects with a robotic arm [8]. The colour recognition is made using image recognition with a webcam. Other examples of active vision system include [9-11]. An economical robotic arm with both image acquisition and range sensing capabilities for machine vision applications is presented by P.C. Nunnally [10]. Robotic Arm is connected via an interface board to an IBM PC/AT microcomputer, offering maximum flexibility in programming the arm for different tasks. The arm was programmed to scan the immediate environment and attempt to locate and identify the nearest object.

The robotic arms are widely used in the industry, but most of them are used in a PTP (Point To Point) trajectory, the moves are learned previously by the robotic arm [3, 14]. Very few robots in the industry are programmed to be smart, or to make decisions. In the future to completely replace the humans with robots, we need robotic arms which can make decisions. This paper proposes the development of a vision-based of controlling an educational robotic arm which can perform color detection [3, 13] and object manipulation [16] together.

This paper is structured as follows. Section 2 introduces the proposed robotic platform. The important materials used and developed methods are discussed in Section 3. In Section 4, an image-processing routine including filtering for the proposed platform are introduced. . Section 5 presents results of the applications developed to demonstrate the platform capabilities. Finally Section 6 comments on conclusion and elaborates the potential of the proposed platform.

978-1-4799-6986-9/14/$31.00©2014 IEEE

II. PLATFORM DESCRIPTION

Figure 1 shows the experimental setup of Educational Robotic Arm (ERA). The overall control system includes an ERA, two Webcams (i.e., WEB1-Top view, WEB2-Side view) and one PC. An ERA includes six metal gear servo motors with 15Kg/cm torque, one microcontroller and its necessary external circuitry. Each Servo control is done by sending each servo a PWM (pulse width modulation) signal, a series of repeating pulses of variable width. The proposed system is a 2D-Image based vision system which is equipped with an intelligent image analysis and object detection algorithm that was developed in MATLAB®. The platform is equipped with resistive force sensor fixed at the robot's end-gripper which facilitates to differentiate objects having different stiffness. A difference here is that two extra rectangular landmarks with red color are used on the ERA (see Fig 5 and Fig 6) read as marker1 (R1) and marker2 (R2) and the objects are marked with green color. The corresponding inputs to PC are the pair of position errors of ERA with respect to the object image centre (i.e., (I_{1ex} ,I_{1ey}) and (I_{2ex} ,I_{2ey}) , the details can be found in Section 4 and the outputs are $\Delta\theta x$ and $\Delta\theta y$ for corresponding servo, denoting the incremental angle change of the ERA platform. The algorithms, including image-processing algorithm for WEB1 and WEB2, and the routines for receiving the image information and sending the reference command, are implemented in the PC.

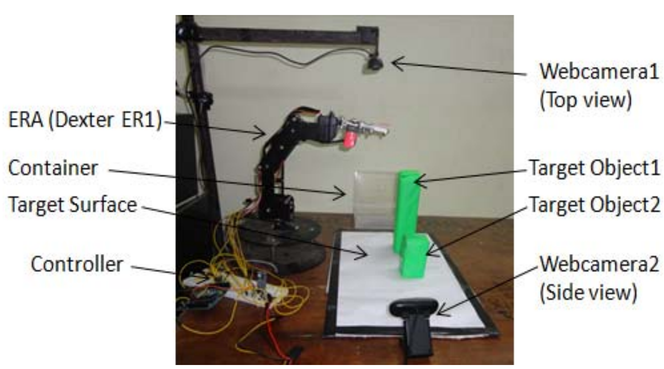

Fig. 1. Image of the automation cell used in this paper.

The proposed framework combines control and image-processing to perform desired operations. In a typical application, once a user's command is encountered, the workspace is scanned and corresponding images are captured. These images are processed to identify target object and marker 1(R1) and marker 2 (R2) for their coordinates. The acquired coordinates are then passed to the PC (Matlab Program) to compute the joint angles required trajectory to reach the target position. Provided the computed angles fall in Range of Motion (ROM) of the arm joints.

III. MATERIALS AND METHODS

The most important elements are two HD 720p web cameras, a light source, an educational robotic arm, a target area defined with a big black line, two light green target cube and a destination container.

A. Educational Robotic Arm

The robotic arm used in this paper is the *Dexter ER-1 Heavy Duty Robotic Arm* (see Fig 2) whose technical specifications and operational data are available in [17]. The educational robot weighs 1.5 Kg with a theoretical load capacity of 50 g. The first motor drives the waist, the second the shoulder, the third the elbow, the fourth and fifth the wrist, and the sixth, the opening and closing of the gripper (see Fig 2).

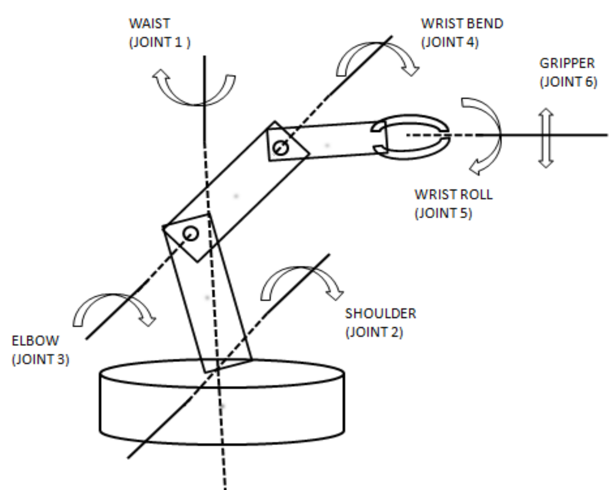

Fig. 2. Schematic representation of educational robot Dexter ER-1.

B. Web Camera

The camera can operate in the RGB format with 24-bit color depth, and in the I420 format with 16-bit color depth. In both formats, the resolution of the images can be 160×120 or 176×144 (non-interpolated images), or 320×240, 352×288 or 640×480 pixels (interpolated images). The default image format and size of this camera is RGB, 24-bit color depth and 320×240 pixels.

C. Color Coded Objects

The development requires the visual detection and identification of two light green cubes (40×40×72 mm, 10 g) and (40×50×170mm, 15 g). Two objects of different dimension are used to show the flexibility of the system. The destination container is currently fixed in the automation cell and not visible to the camera.

D. Object Type Detection

Before the control operation of the robotic arm, the object properties are examined such as its shape andits colour. In this paper, the algorithm is such that the robotic arm is controlled only if the object is green coloured and has a square shape. MATLAB function regionprops() can automatically calculate some properties of the input image, such as the major axis length, minor axis length and extent.

The value extent is calculated by the rate of minimum enclosing rectangle's area of the image and the image's area. The minimum enclosing rectangle for square and rectangle are themselves, therefore the extent values for rectangle and square are both 1. However, these values of major axis length and minor axis are different for rectangle.

Figure 3 shows the RGB values for about 100 pixels of the target object. This helps us to identify the colour of the object. Figure 4 brings out the summary of the total robotic arm operation.

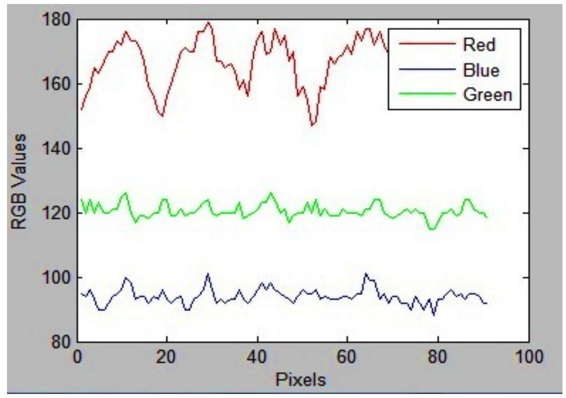

Fig. 3. Plot to show the RGB values for the object.

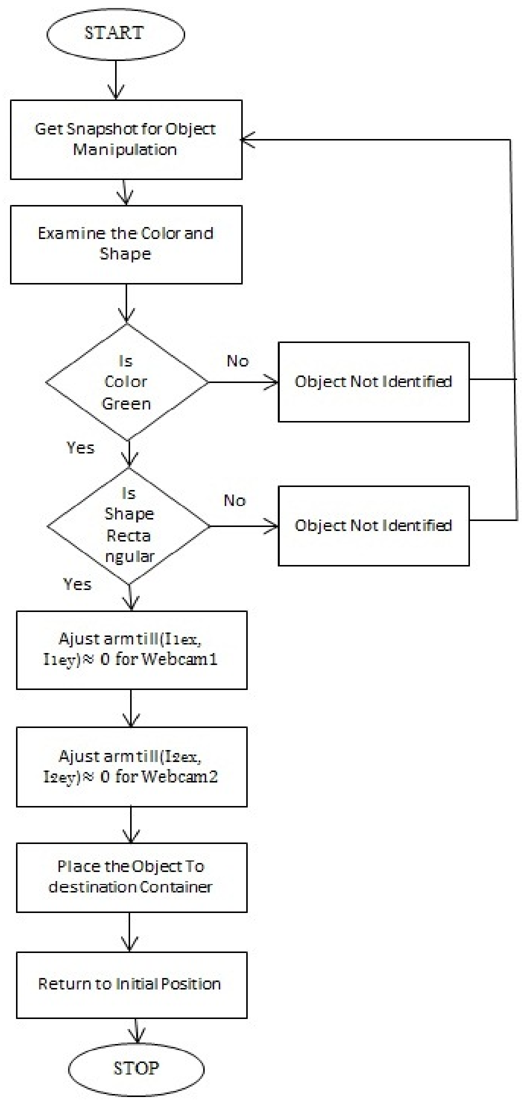

Fig. 4. Flow chart diagram of Robotic arm Operation.

IV. IMAGE ACQUISITION AND PROCESSING

A. *Image acquisition*

This intermediate block was focused on describing and applying standard signal processing techniques in images [26]. These techniques cover image enhancement algorithms such as noise reduction, filtering, contrast adaptation, etc., and also image analysis procedures applied after segmentation and morphological filtering, such as size, position, orientation, distance and average color estimate. The Matlab environment has enormous library of functions (or toolbox) dedicated to "*Image Acquisition*". The image acquisition toolbox can be used to access image acquisition devices and store a sequence of images for offline analysis. The destination of the images acquired can be set as logging to disk or as logging to memory.

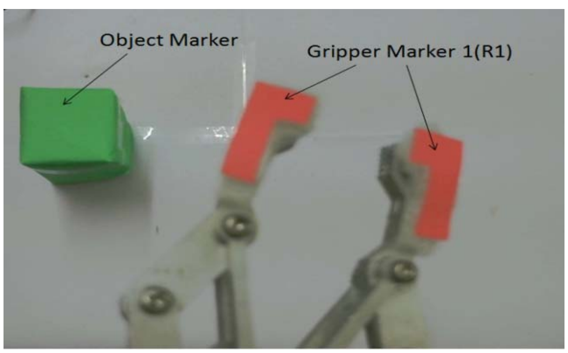

Fig. 5. Image captured by Webcamera1

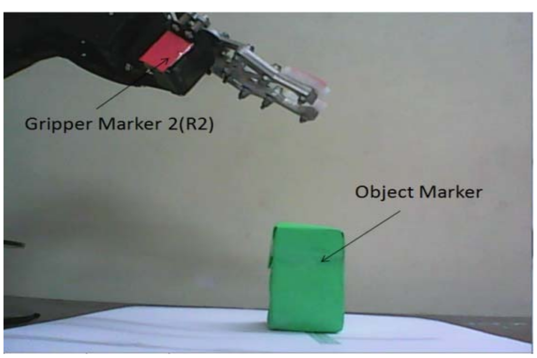

Fig. 6. Image captured by Webcamera2

B. Conversion of Color Space

Most digital images use the *RGB* color space. However, individual *R*, *G*, and *B* components may have unstable variations under changing illumination conditions. On the other hand, it is easier to use the *Y UV* color space (where *Y* represents the luminance component, and *U*, *V* are chrominance components) for the segmentation of desired features. Vadakkepat *et al.* [23] verified that the *UV* color space for the face tracking problem is more effective and robust than that of the RGB. Hence, the color base of *RGB* is first transformed into that of *Y UV* with *U* and *V* ∈ [0, 255] as

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & -0.500 \\ 0.5 & -0.419 & -0.018 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \tag{1}$$

C. Image Segmentation

Since the value of *Y* is strongly dependent on luminance, it will not be considered in this paper. The values of *U* and *V* are set to different ranges as $U \in [U\text{min}, U\text{max}]$ and $V \in [V\text{min}, V\text{max}]$ for the identification of different objects, e.g., marker1 (R1), marker2 (R2) and object in Fig 5 and 6. Hence, the ranges [*U*min, *U*max] and [*V*min, *V*max] for the image features of R1 and R2, i.e., rectangular landmarks with red color, are assigned as [0, 110] and [0, 120], respectively. Similarly, the image features of the rectangular object with light green color, i.e., [*U*min, *U*max] and [*V*min, *V*max], are set as [150, 255] and [120, 255], respectively. They are, respectively, on the third and first quadrants of *U* and *V*. The process of image segmentation is more robust in distinguishing marker R1 and R2 on the ERA and the object on the ground with non uniform illumination, and strong reflection.

D. Binary

The type of image processing followed is real-time processing, which has fast image processing time and is immune to the varying of the size irrespective of how far the object is from the camera. Color filtering technique is used to extract the current position of the gripper and the static position of the object. The object of interest colors to be considered is red for the current position of the gripper and light green color for the object while other colors are discarded. The binary of the image is to choose a suitable threshold value *Tb* for *U* and *V*. The corresponding relation is

$$P\,(Ix, Iy) = \begin{cases} 1 \ \text{if} \ f(Ix, Iy) \ \geq \ Tb \\ 0 \ \text{if} \ f(Ix, Iy) < \ Tb \end{cases} \qquad (2)$$

where f(*Ix, Iy*) denotes the values of *U* and *V* on the image plane (*Ix, Iy*), *P*(*Ix, Iy*) = 1 stands for the white pixels, and *P*(*Ix, Iy*) = 0 stands for the black pixels. The purpose of the binary is to reduce the storage amount as well as the computation load. The value of *Tb* is less sensitive to lighting conditions because the *Y* component is not considered for the binary operation. In Fig 7 and Fig 8 the green color object and the two red marks R1 and R2 on the ERA are segmented from the background for further processing.

E. Filtering

A median filtering is used to remove noises produced by additional factors such as lighting intensities and the presence of moving particles. Median filtering is similar to using an averaging filter, in that each output pixel is set to an average of the pixel values in the neighbourhood of the corresponding input pixel. However, with median filtering, the value of an output pixel is determined by the *median* of the neighbourhood pixels, rather than the mean. The median is much less sensitive than the mean to extreme values (called *outliers*). Median filtering is therefore better able to remove these outliers without reducing the sharpness of the image. Then a Gaussian filter is used to further smoothen the image but will preserve edges better than the more basic mean filter. The resulting object is shown in Fig 9 and Fig 10 for Webcam1 and Webcam2**.** By weighting a pixels contribution to the final pixel value this filter can better preserve edges than the mean filter which specifies equal weights to all pixels within the filter window. For a 1-D Gaussian filter the single filter values are defined as

$$G(x) = \sqrt{\frac{1}{\sqrt{2\pi}}}\ .e^{\frac{x^2}{2^2}} \qquad (3)$$

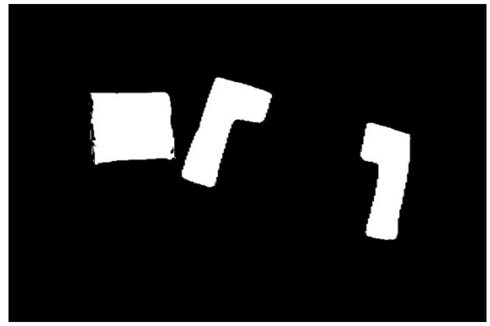

Fig. 7. Binary segmented image for Webcam1

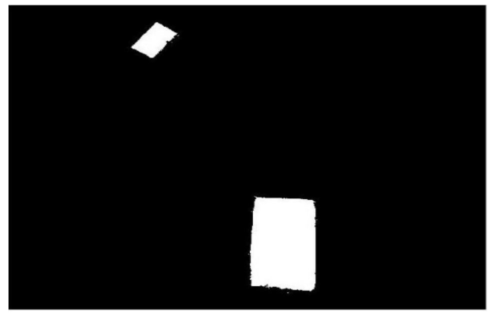

Fig. 8. Binary segmented image for Webcam2.

Fig. 9. Noise reduced and filtered image for Webcam1.

Fig. 10. Noise reduced and filtered image for Webcam2.

F. Image Representation and Description

The centre and coordinate of the image features are considered to represent the pose of the ERA on the image plane coordinate. For the digital image, their 2-D centred moments are defined as follows [22]–[24]:

$$(I_{1ix}, I_{1iy}) = \sum_{(I1x,I1y)\in\Omega i} \sum (I1x, I1y)\,/N \qquad (4)$$

$$(I_{1ox}, I_{1oy}) = \sum_{(I1x,I1y)\in\Omega o} \sum (I1x, I1y)/N \qquad (5)$$

The centres of the image object1 and image gripper1 i.e. (I_{1ix}, I_{1iy}) and (I_{1ox}, I_{1oy}) in fig 11 can be calculated by (4) and (5) respectively for the desired image features Ωi and Ωo. Similarly, the centres for the image object2 and image gripper2 i.e. (I_{2ix}, I_{2iy}) and (I_{2ox}, I_{2oy}) in fig 12 are found. The error between centres of the image object1 and image gripper1 and also between the image object2 and image gripper2 are determined as:

$$(I_{1ex}, I_{1ey}) = (\,|\,I_{1ix} - I_{1ox}\,|\,,\ |\,I_{1iy} - I_{1oy}\,|\,) \qquad (6)$$

$$(I_{2ex}, I_{2ey}) = (\,|\,I_{2ix} - I_{2ox}\,|\,,\ |\,I_{2iy} - I_{2oy}\,|\,) \qquad (7)$$

V. EXPERIMENTAL RESULTS

The control of the individual servo direction and orientation is based on the principle of the difference between the centre of the gripper image and the object image. The equations (6) and (7) helps in determining the error distances (I_{1ex}, I_{1ey}) and (I_{2ex}, I_{2ey}) for the images obtained from WEB1 and WEB2. The initial step was the movement of ERA to reduce the x-direction error I_{1ex} by an incremental step size angle of ±2˚ (degrees). After which theWEB2 image is analysed to lower the gripper position at the centre of the object. On successful completion of each iteration of the image processing, one byte of data is send to the predefined COM Port to which microcontroller configured as serial receiver is connected. Depending on the data byte received in the UART buffer, microcontroller operates the ERA robot in specified direction. When there is no byte present inthe UART buffer of the microcontroller, it continuously waits for the next data. After the proper orientation of the gripper, the object is lifted up and moved to a predefined destination.

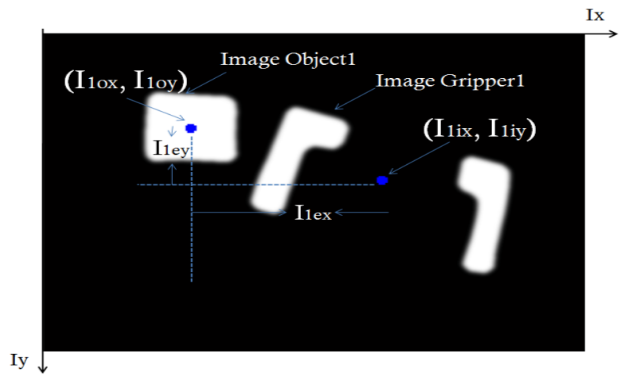

Fig. 11. Image representing gripper and object centre coordinates for Webcam1.

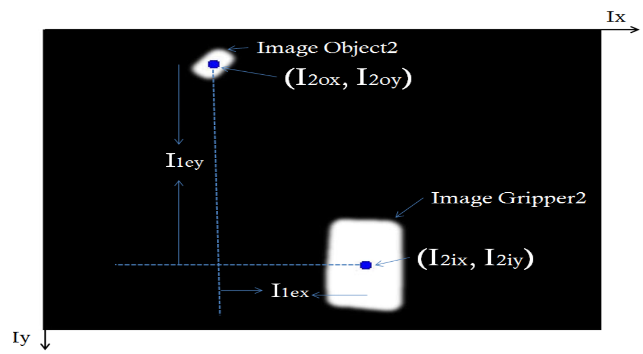

Fig. 12. Image representing gripper and object centre coordinates for Webcam2.

Corresponding to the pick and place operation for an object using the actual platform, Fig 13 and Fig 14 illustrates trajectories followed by various joints of the robot. The Position data of the six motors has been plotted in a sequence of the robot moving to pick the object, closing the gripper to grasp the object, moving towards destination position, opening the gripper to release the object and finally to return to its home position.

The difference in the trajectories of robot joints while picking up and placing the target object1 and target object2 can be noticed in the degrees covered by joint3 and joint 2. The angular movement of joint 3 and joint2 is more in case of object2 compared to object1 (see Fig 13 and Fig 14). Varying the object and also its destination coordinates in case of pick and place operation, recording the positions (in both x and y) followed by the robot can give an estimate of the positional accuracy of the robot and its application.

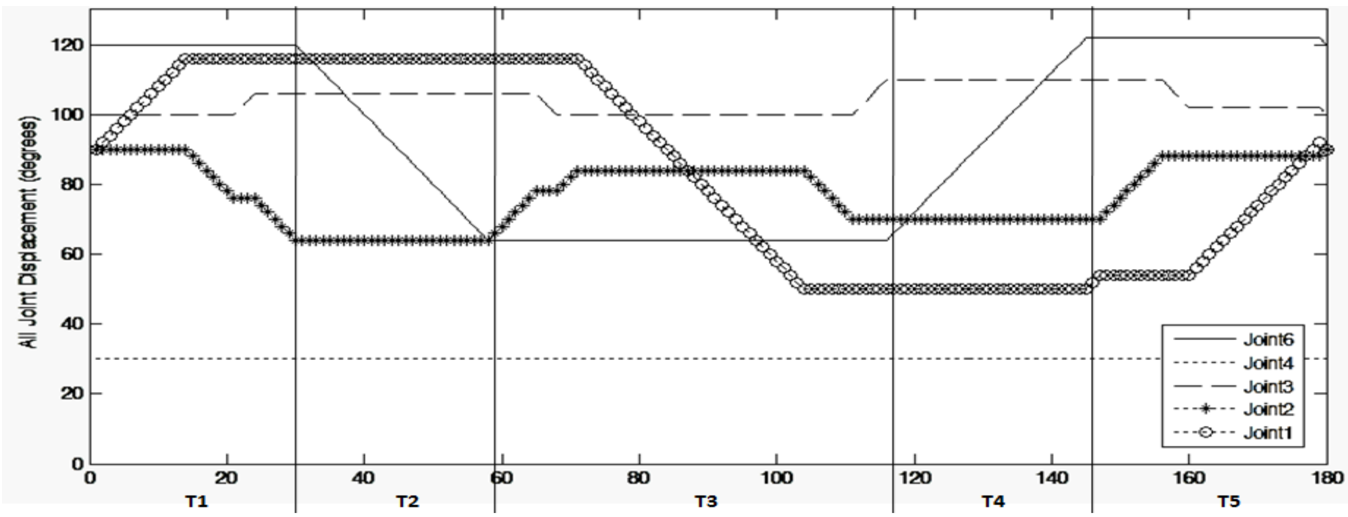

Fig. 13. Trajectories of robot joints while picking up and placing the target object1.

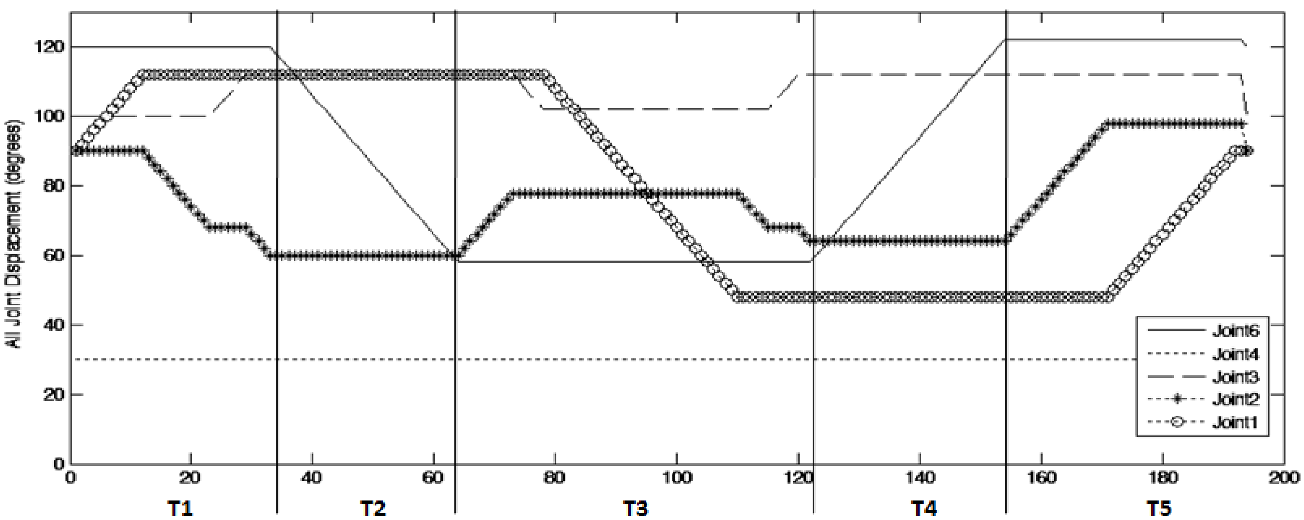

Fig. 14. Trajectories of robot joints while picking up and placing the target object2.

T1: Moving to pick up the target T2: Closing the gripper T3: Carrying the object to the Destination T4: Opening the Gripper T5: Going back to the home position.

VI. CONCLUSION

A methodology for the trajectory tracking of ERA using two Web cameras has been developed. The framework of an educational environment to practically grasp using object manipulation has been presented in this research. Promising features of the platform include availability of multiple sensory feedbacks, 2D visualisation, higher level commands and cost effective. This paper has successfully demonstrated the application of an intelligent robotic arm control system using computer vision. The proposition of the control scheme is to achieve good tracking performance with the flexibility of easy object manipulation method. The design procedure of the proposed optimal control system was described in detail. Moreover, experimentation was carried out using different reference trajectories to verify the effectiveness of the proposed control system.

The implemented experiments based on the proposed platform demonstrate its relevance with academic courses like Robotics, Control systems, Mechatronics, Image processing, etc. The proposed system is equally good for object handling and manipulation in an industrial environment for applications like welding, CNC machine tending, machine vision and various Flexible Manufacturing System (FMS) tasks. The scope of this platform has the capacity to get enhanced by simple replacing the robot's gripper with other tools like paint brush, drill, grinding wheel, etc.

REFERENCES

[1] Albert M. Cook*, Senior Member, IEEE*, Brenda Bentz, Norma Harbottle, Cheryl Lynch*, Student Member, IEEE*, and Brad Miller "School-Based Use of a Robotic Arm System by Children With Disabilities" IEEE transactions on Neural Systems and Rehabilitation Engineering, vol. 13, no. 4, December 2005.

[2] Sarah Manzoor, Raza Ul Islam, Aayman Khalid, Abdul Samad, Jamshed Iqbal "An open-source multi-DOF articulated robotic educational platform for autonomous object manipulation" Robotics and Computer-Integrated Manufacturing 30 (2014) 351–362

[3]Tomàs Pallejà Cabré Jordi Palacín Roca, "Project-based learning example: controlling an educational robotic arm with computer vision," IEEE Revista Iberoamericana De Tecnologias Del Aprendizaje, vol. 8, no. 3, August 2013

[4] Shaikh, A. ; M.H. Saboo Siddik, Khaladkar G. ; Jage, R. ; Taili T.P.J.

Robotic Arm Movements Wirelessly Synchronized with Human Arm Movements Using Real Time Image Processing India Educators' Conference (TIIEC), 2013 Texas Instruments.

[5] Dragusu, M. Mihalache, A.N. ; Solea, R. Practical applications for robotic arms using image processing System Theory, Control and Computing (ICSTCC), 2012 16th International Conference .

[6] W. I. Clement and K. A. Knowles, "An instructional robotics and machine vision laboratory," *IEEE Trans. Educ.*, vol. 37, no. 1, pp. 87–90, Feb. 1994.

[7] C. Solomon and T. Breckon, Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab. New York, NY, USA: Wiley, 2010.

[8] Szabo, R. Lie, I. Automated colored object sorting application for robotic arms Electronics and Telecommunications (ISETC), 2012 10th International Symposium .

[9] Christoforou, E.G. ; Dept. of Radiol., Washington Univ., St. Louis, MO ; Ozcan, A. ; Tsekos, N.V. Robotic Arm for Magnetic Resonance Imaging Guided Interventions Biomedical Robotics and Biomechatronics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference.

[10] Nunnally, P.C. Weiss, J.M. Tsay, T.I.J. Lee, M.K. ; Wang, G.L. ; Hsu, M.S. An inexpensive robot arm for computer vision applications Southeastcon '89. Proceedings. Energy and Information Technologies in the Southeast., IEEE.

[11] D. Kragic, H.I. Christensen, Survey on visual servoing for manipulation, Royal Institute of Technology (KTH), Stockholm, Sweden, Tech. Rep. ISRN KTH/NA/P-02/01-SE, CVAP259, 2002.

[12] F. Chaumette, S. Hutchinson, Visual servo control, II, advanced approaches, IEEE Robotics Automation Magazine 14 (1) (2007) 109 118.

[13] Manseur R. Visualization tools for robotics education. In: Proceedings of the international conference in engineering education (ICEE) 2004.

[14] A. Emin and G. Erdemir, "Development of a web-based control and robotic applications laboratory for control engineering education," *Inf.Technol. Control*, vol. 40, no. 4, pp. 352–357, 2011.

[15] N. Vahrenkamp, C. Boge, K. Welke, T. Asfour, J. Walter, R. Dillmann, Visual servoing for dual arm motions on a humanoid robot, in: IEEE International Conference on Humanoid Robots, 2009, pp. 208 214.

[16] Fei-Yue Wang, Paul J.A Lever and Bing Pu, “A Robotic Vision System for Object Identification and Manipulation Using Synergetic Pattern Recognition”, Robotics & Computer-Integrated Manufacturing 10, No. 6, pp. 445-59, 1993.

[17] Web of the Robotic Arm Dexter ER-1 [Online]. Available:

http://www.nexrobotics.com/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=6

67&category_id=124&option=com_virtuemart&Itemid=45

[18] V. Lippiello, B. Siciliano, L. Villani, An experimental setup for visual servoing applications on an industrial robotic cell, in: IEEE/ASME International onference on Advanced Intelligent Mechatronics, July 2005, pp. 1431 1436.

[19] C. Solomon and T. Breckon.: *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*.,New York, NY, USA: Wiley, 2010.

[20] R. Kelly, R. Carelli, O. Nasisi, B. Kuchen, F. Reyes, Stable visual servoing of camera-in-hand robotic systems,

IEEE/ASME Transactions on Mechatronics 5 (1) (2000) 39 48.

[21] J. Zhou, Y. Yu, Coordination control of dual-arm modular robot based on position feedback using Optotrak3020, Industrial Robot: An International Journal 38 (2) (2011) 172 185.

[22] E. D. Davies, *Machine Vision—Theory, Algorithms, Practicalities*, 3rd ed.Amsterdam, The Netherlands: Elsevier, 2005.

[23] P. Vadakkepat, P. Lim, L. C. De Silva, L. Jing, and L. L. Ling, “Multimodal approach to human-face detection and tracking,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 3, pp. 1385–1393, Mar. 2008.

[24] T. Komuro and M. Ishikawa, “A moment-based 3D object tracking algorithm for high-speed vision,” in *Proc. Int. Conf. Robot. Autom.*, Rome, Italy, Apr. 10–14, 2007, pp. 58–63.

[25] A. Emin and G. Erdemir, “Development of a web-based control and robotic applications laboratory for control engineering education,” *Inf Technol. Control*, vol. 40, no. 4, pp. 352–357, 2011

[26] C. Solomon and T. Breckon.: *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*.,New York, NY, USA: Wiley, 2010.