Pose Estimation of Top Object on Stack of Elliptical Objects Based on Ellipse Fitting and Kalman Estimator

Sangsoo Noh, Sangbum Park, Youngjoon Han, Hernsoo Hahn
School of Electronic Engineering, Soongsil University
1 Sangdo-Dong, Dongjak-Ku, Seoul, 156-743, Korea
http://visionlab.ssu.ac.kr

Abstract: - This paper presents a top object selection and its pose determination algorithm which can be used to pick the top one among a stack of overlapped elliptical objects provided on a moving conveyer belt. For this purpose, an ellipse fitting algorithm is developed to find the ellipse shapes and their parameters in the image. It also determines the top object among them and its pose by analyzing the edge patterns. Based on the object's speed and pose measured in the image frame, the Kalman estimator determines the pose of the object at the moment of pick-up with considering the noises caused by the motions of robot and conveyer belt. Once the object's pose to pick is determined in the image frame, then its corresponding robot's pose in the world frame is determined using the image Jacobian. The proposed algorithm implemented and installed in the 6DOF Denso robot has shown a prominent performance in the experiments.

Key-Words: - Top object detection, Bin picking, Overlapped objects in a moving conveyer belt, Ellipse fitting, Pose estimation.

1 Introduction

Recently, vision based robot control techniques have been actively introduced for various applications in industries, such as tracking a moving object and sorting parts on a moving conveyer belt. These techniques make a great contribution in implementing flexible manufacturing systems by enhancing the intelligence of robots while decreasing the complexity involved in the implementation of manufacturing machines.

The most popular example of vision based control system can be found in a bin picking problem. [1] The conventional part feeding systems are asking the objects to be located in the bins located at the pre-specified position with known sizes. This constraint restricts significantly manufacturing systems from being used for other purposes. Such restrictions imposed on the position and size of the objects can be removed by a vision based control system. For example, the vision system developed by Fanuc [2] allowed a robot to pick up the top object among the arbitrarily stacked objects in a basket.

Implementation of such system includes many research problems such as top object detection and localization, prediction of object's pose to pick up, transformation of the image frame to the world frame, and synchronization (real-time communication) of vision and robot processes, etc. Among these, the most fundamental problems are how to select the object to pick up and how to predict its pose in terms of the world frame at the moment of pick-up. Since the most objects in industries include elliptical shapes in them, this paper deals with only the elliptical (including circular) objects.

For detecting the elliptical objects in an image, model based approaches are most popularly used since it is simple and fast if the object's size is fixed. For example, Luo [3] used as the models the feature points on independent view-points of the ellipses which can be derived from a modified 3-D General Hough Transformation. Since the accuracy of detection operation depends on the number of feature points, it is pointed as a weakness of this approach that the computational complexity increases to enhance the accuracy.

Randomized Hough transformation is also a good solution for detection of elliptical objects. [4] This method represents an ellipse by three parameters which can be extracted if three points of the ellipse are given. Thus, the ellipse parameters of every three edge points are extracted and those points are merged as one ellipse if they have the same ellipse parameters. Although these methods have shown good performance when the objects are located separate or

small number of objects exists in the image, they produce frequently false ellipses when multiple objects are overlapped and require a long processing time.

Among the approaches proposed for solving another problem of predicting the pose of the object at the moment of pick-up, some have shown good performance. For example, Kosmopoulos and Psomopoulos [5] have made the database of the robot's pose corresponding to the patterns of the object's features in a 2D image, using the image Jacobian. This method has an advantage that it is not dependent on the depth measurement and calibration.

These methods are effectively used when the camera does not move, but their performance cannot be maintained when both the camera and objects move simultaneously.

To solve these problems, this paper proposes a top object detection algorithm to pick based on ellipse fitting scheme and a pick-up pose prediction algorithm based on velocity prediction scheme using Kalman filter. The proposed ellipse fitting algorithm groups the boundary points of an ellipse by the line segments and then finds the ellipse parameters including the ellipse center. From the ellipse parameters, the pose of the object is calculated. When objects are overlapped, then the top object is selected by testing the edge types of individual ellipses. Once the object's pose in the current image is determined, its pose to pick in the next stage is estimated by a Kalman estimator, and its corresponding robot's pose in the world frame is calculated using the image Jacobian.

Fig. 1 shows the configuration of the proposed system. In the first step, the camera located at the end-effecter captures the objects' image on the conveyer belt and then the top object to pick and its pose in the next stage are determined. The robot moves to the pick-up position and confirms that the object is aligned with the robot's end-effecter. This procedure is repeated until the object's axis and the end-effecter are aligned, as shown in the Fig. 1.

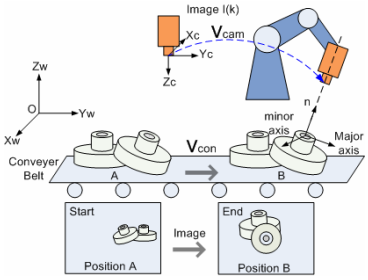


Fig. 1 System Overview

2 Ellipse Fitting Algorithm

From the input image, the object to pick that is located on the top among a stack of objects should be determined first. For this purpose, the sequence of processes given in Fig. 2 is executed.

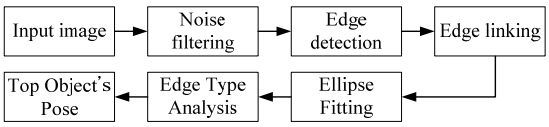


Fig. 2 The flow of the top object's pose determination algorithm.

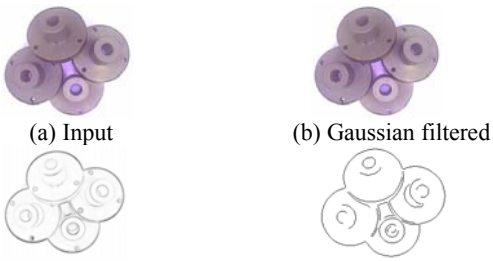
2.1 Image Processing

Since the main process of the algorithm to detect objects uses an edge image, the algorithm begins with an edge detection process. To reduce a noisy illumination effect in capturing an image, a special lighting device is designed as shown in Fig. 3(a). It has multiple lights around the camera, each of which can be individually controlled to compensate the light conditions of the working area so that most clear edge image can be obtained. The image obtained using the lighting device and given in Fig. 3(c) proves the positive effects of the lighting device.

(a) Lighting system (b) Image without (a) (c) Image with (a)

Fig. 3 The lighting system and an acquired image compared to the one obtained without the light system

To remove the noise component still included in the input image, the Gaussian filter is used since the motions of robot or users under fluorescent lamp may generate high frequency noise in the image. Then, the edges are detected using the Sobel operator followed by a thinning operation. The Sobel operator is used since it does not miss edges although it produces thick ones. Fig. 4 shows the images obtained in this preprocessing stage.



(a) Input (b) Gaussian filtered

(c) Sobel Edge (d) After thinning

Fig. 4 Images in each stop of the preprocessing stage

2.2 Ellipse Detection

For accurately detecting all ellipses in the input image, the ellipse fitting algorithm based on RHT (Randomized Hough Transformation) is used. RHT uses three edge points (*X1*,*X2*,*X3*) to derive the three ellipse parameters (*A*,*H*,*B*) of Eq. (1), as shown if Fig. 5. Eq. (2) is the ellipse condition that the ellipse parameters should satisfy.

$$Ax^2 + 2Hxy + By^2 = 1 \quad (1)$$

$$AB\text{-}H^2 > 0 \quad (2)$$

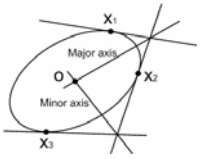
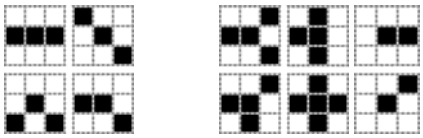


Fig. 5 Ellipse center derivation from three edge points in RHT

Since the RHT has a problem of detecting false ellipses if ellipses are overlapped, here the ellipse fitting algorithm is used which uses edge segments, instead of using edge pixels. It detects the edge segments by linking of edge points. To find the edge segments, an edge pixel is selected in the image and its neighboring edge pixels in an 8-connectivity window are registered. The registered edge pixels are considered as the same edge segment if they have only two neighboring edge pixels as shown in Fig. 6(a). If an edge connected in the window has more than 3 or less than 2, it is considered as the end point of the edge segment as shown in Fig. 6(b)



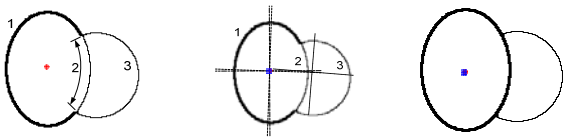
(a) Merging patterns (b) Ending patterns

Fig. 6 Procedure of extracting the edge segments

Once the edge segments are labeled, then their ellipse parameters are derived if an edge segment satisfies the ellipse condition given in Eq. (2). Then the edge fitting process tests the ellipse parameters of the edge segments, pair by pair, to merge them as one ellipse if they have the same ellipse parameter. Eq. (3) is used as the merging condition of the edge segment pair.

$$C = N_{A/B}\, N_{B/A} > k \quad (3)$$

In Eq., (3), $N_{A/B}$ is the ratio of the pixels of line segment A which can be pertained to the ellipse generated by line segment B. Thus *C* in Eq. (3) shows the possibility that two line segments can be pertained to the same ellipse. *k* is the threshold to be determined by experiments. Fig. 7(a) shows the edge segment image where each edge segment is labeled. Fig. 7(b) illustrates graphically the ellipse parameters of individual line segments and Fig. 7(c) shows the ellipse image finally acquired by the above merging process.



(a) Edge segments (b) Ellipse parameters (c) Detected ellipses

Fig. 7 Detection of ellipses from edge segments

2.3 Determination of the top object and its pose

Based on the image where all ellipses are detected, the top one is searched by analyzing the edge types included in the ellipses. If the boundary of an ellipse has no any branching point, then the ellipse is separated from others and thus can be considered as a top object. Otherwise, it is overlapped in any sense with other ellipses. If an ellipse has an edge segment whose end point has a branch as shown in Fig. 8, then it should be tested whether it is an occluding or occluded one.

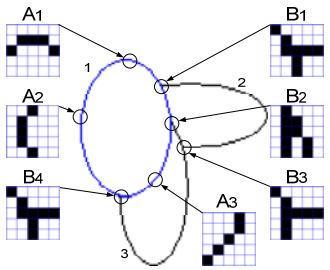


Fig. 8 Edge patterns of 'occluding' and 'occluded'

In Fig. 8, the patterns labeled by A_n can be found in non overlapping edges and has no branch. The patterns labeled by B_n are found in overlapping edges and characterized by T type branches in them. In the patterns labeled by B_n, the edges branching out are the occluded ones and the main streaming edges are the occluding ones. Thus, if an ellipse includes even an occluded edge at the junctions, it is an occluded ellipse. In Fig. 8, since ellipses 2 and 3 have more than one occluded edges, they are considered as the occluded ellipses. Since ellipse 1 does not include any occluded edges, it is considered as the top ellipse.

3 Determination of the Ellipse Pose To Pick

3.1 Kalman filter to estimation the object's pose to pick in the image frame

In the world frame, since the conveyer belt is moving at a constant speed v_c, the position p_n of the object at the moment of pick-up t_n can be calculated by adding the distance d_m which is a multiplication of the v_c and the sampling period t_s to the current position of the object p_c, as shown in Eq. (4).

$$p_n = p_c + d_m = p_c + v_c \times t_n \tag{4}$$

However, it cannot be used as it is due to the noises included in the measurement and caused by the motions of robot and conveyer belt. As can be found in Fig. 9, the speed of the object V_{obj} in the image frame is the summation of the speed of the conveyer belt V_{con} and the speed of the robot (camera) V_{cam}, as shown in Fig. 9, both of which include noises. To take into account these noises and to accurately determine the object's pose at the moment of pick-up, a Kalman estimator is used in this paper.

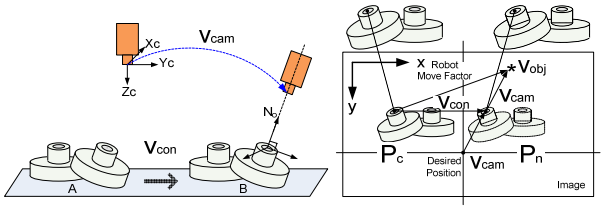


Fig. 9 Speed of object in the image

If the current position of the object to pick in the image is P_c, then the position at the moment of pick-up P_n can be estimated using a Kalman estimator, given in Eq. (5). Eq. (5a) is the state transition equation where $s(k)=[x(k),y(k),x'(k),y'(k)]^T$ is the state vector, and Eq. (5b) is the measurement equation where $z(k)=[x(k),y(k)]^T$.

$$s(k)=As(k-1)+n(k) \tag{5a}$$

$$z(k)=Hs(k)+e(k) \tag{5b}$$

In Eq. (5a), A is the transition matrix and *n(k)* is the noise illustrating the unexpected effects generated during the motion of the robot and conveyer belt. In Eq. (5b), H is the measurement matrix and *e(k)* is the noise caused by camera motion and illumination conditions. Thus, *n(k)* and *e(k)* can be modeled by Gaussian functions. Using the statistical features of the vectors in Eq. (5), the prediction and correction of the state are determined by Eq. (6) and Eq. (7)

$$P^-(k)=AP(k-1)A^T+Q \tag{6a}$$

$$\hat{s}^-(k)=A\hat{s}(k-1) \tag{6b}$$

$$K(k)=P^-(k)H^T[HP^-(k)H^T+R]^{-1} \tag{7a}$$

$$P(k)=[I-K(k)H]P^-(k) \tag{7b}$$

$$\hat{s}(k)=\hat{s}^-(k)+K(k)[z(k)-H\hat{s}^-(k)] \tag{7c}$$

The first measurements of the object's position and speed are given as the initial state, and the object's position is recursively calculated by the above Kalman estimator.

3.2 Image Jacobian to determine the robot's pose to pick the object

To pick the object at the position estimated by the Kalman filter, the configuration of the robot is determined by using the image Jacobian. Since the goal of the system is to pick the top object, its pose is given as the constraint in the calculation of the robot's pose. That is, the robot's approaching axis (Z axis of the end-effector in this case) should coincide with the elliptical object's center axis.

The image Jacobian shows the relationship between the differential motion of the camera and the differential motions of robot joints. Since the camera's motion can be derived from the object's motion in the image, the camera's motion can be interpreted as the object's motion. Then, the differential motion of the camera in the Jacobian can be replaced by that of the object, which is represented by Eq. (8).

$$\dot{P}_c=[\dot{X}_c,\dot{Y}_c,\dot{\omega}]^T \tag{8}$$

where *w* is the scale facetor.

To achieve this motion of the object in the world plane, the differential motions of individual joints of the 6DOF Denso robot can be represented by the following vector, given in Eq. (9).

$$u=\frac{\Delta\theta_{(n-c,i)}}{\Delta t}=\left[\dot{\theta}_1,\dot{\theta}_2,\dot{\theta}_3,\dot{\theta}_4,\dot{\theta}_5,\dot{\theta}_6\right]^T \tag{9}$$

Then, the image Jacobian relating these two vectors is reprented by Eq. (10).

$$\dot{P}=J_{Image}u \tag{10}$$

To derive J_{image}, the relationship between the position of the object in the image frame P_c and that in the world frame p_c should be known first. Their

relationship in the case where the object is not moving is represented by Eq. (11) and that in the case where the object in motion is represented in Eq. (12).

$$P_c = Ap_c \tag{11}$$
$$\dot{P}_c = A\Delta p_c \tag{12}$$

By inserting Eq. (8) to Eq. (12), Eq. (12) becomes as floows.

$$\dot{P}_c = A\Delta p_c$$
$$\begin{bmatrix} \dot{X}_c\omega + X_c\dot{\omega} \\ \dot{Y}_c\omega + Y_c\dot{\omega} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \alpha_x f & 0 & 0 & 0 \\ 0 & \alpha_y f & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & -\delta_z & \delta_y & d_x \\ \delta_z & 0 & -\delta_x & d_y \\ -\delta_y & \delta_x & 0 & d_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \tag{13}$$

where Δ is the homegeneous matrix showing the transformation between the image frame and the world frame and A is the camera projection matrix. Now Eq. (13) can be rearranged in terms of $[d_x, d_y, d_z, \delta_x, \delta_y, \delta_z]^T$ by replacing w by Z_c and $\dot{\omega} = \delta_x y_{obj} - \delta_y x_{obj} + d_z$ to obtain the following Jacobian matrix given in Eq. (14).

$$J_{Image} = \begin{bmatrix} \frac{\alpha_x f}{f - z_n} & 0 & \frac{\alpha_x f x_n}{(f - z_n)^2} & \frac{\alpha_x f x_n y_n}{(f - z_n)^2} & \frac{\alpha_x f}{f - z_n}\left(z_n - \frac{(x_n)^2}{f - z_n}\right) & \frac{\alpha_x f y_n}{f - z_n} \\ 0 & \frac{\alpha_y f}{f - z_n} & \frac{\alpha_y f y_n}{(f - z_n)^2} & \frac{\alpha_y f}{f - z_n}\left(-z_n + \frac{(y_n)^2}{f - z_n}\right) & \frac{-\alpha_y f x_n y_n}{(f - z_n)^2} & \frac{\alpha_x f x_n}{f - z_n} \end{bmatrix} \tag{14}$$

where (x_n, y_n, z_n) is the coodinates of the object in the world frame, a_x and a_y are the image scaling factors in X and Y axes, and f is the camera focal length. They are the camera variables which are obtained in the camera calibration process.

To determine the pose of the robot when the object in located at P_n uisng the above image Jacobian, the differential motion of the object in the image plane is replaced by Eq. (15).

$$P = \frac{\Delta P}{\Delta t} = \frac{(P_n - P_c)}{\Delta t} \tag{15}$$

Then the differential motion of the robot joints can be extracted from the image Jacobian $\dot{P} = J_{Image} u$ as Eq. (16).

$$u = J_{Image}^{-1} \dot{P} \tag{16}$$

Since the numbers of elements in $\dot{P}$ and u are not same, J_{Image}^{-1} cannot be simply determined. In this case, the pseudo inverse is derived as shown in Eq. (17) and it is used to determine u as given in Eq. (18).

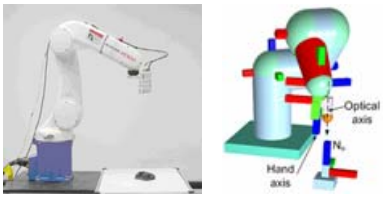
$$J_{Image}^{+} = (J_{Image}^{T} J_{Image})^{-1} J_{Image}^{T} \tag{17}$$
$$u = \lambda J_{Image}^{+} (P_n - P_c) \tag{18}$$

where $\lambda > 0$ is the step size.

4 Experiment

The proposed algorithm has been implemented and installed on a 6DOF Denso Robot. A camera having 320x240 pixels is mounted on the end-effecter in such a way that the approaching axis of the hand and the optical axis of the camera are parallel. Initially, the robot is located so that the optical axis of the camera is perpendicular to the conveyer belt plane to capture the image of the objects whey they arrive at the predetermined position, as shown in Fig. 10(a). The final goal is to locate the robot at the grasping pose as shown in Fig. 10(b) where the approaching axis of the gripper is perpendicular to the plane of the ellipse to pick. The test objects used in the experiments are consisted of two cylinders.

The performance of the system has been evaluated in the three viewpoints. The first one is the success rate of the top object detection among the stack of multiple objects. The second is the accuracy of the pose of the ellipse which is represented by the center position and the long and short axes. The third is the accuracy of the position estimation.



(a) initial pose (b) final pose

Fig. 10 The experimental set using Denso robot

Fig. 11 shows several images used in the experiment where test objects are arbitrarily stacked and the resulting images where the top objects are detected. Total 100 images where 3 to 5 objects are stacked are tested and the proposed algorithm have shown the success rate of 98%.

To measure the pose of the detected object, the ellipse parameters are derived using the ellipse fitting equations given in Eq. (1). The measured poses of the top objects given in Fig. 11 are compared to the actual poses measured manually, and summarized in Table 1. The pose is represented by $(x, y, \theta_x, \theta_y)$ where (x,y) are the coordinates of the ellipse center in pixels and θ_x and θ_y are the rotational angle about the X and Y axes in angle, respectively. As summarized in Table 1, the average measurement error is less than 2 pixels in position and 1.5^o in orientation.

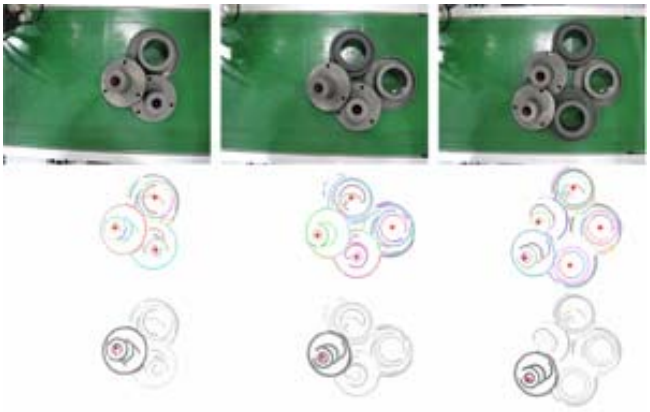


Fig. 11 Top object detection: upper figures are the input images and lower figures are the resulting ones

Table 1. The poses of the top objects given in Fig. 11, measured by the proposed algorithm and the actual ones measured manually

Image Number	Measured	Actual	Measurement Error
1	(141,123,30,32)	(142,124,31,31.5)	(1,1,1,0.5)
2	(102,145,30,45)	(101,144,31,44)	(1,1,1,1)
3	(103,167,23,61)	(103,166,22,62)	(0,1,1,1)

Based on these measurements, the next pose of the objects are estimeated using the Kalman estimator. The accuracy of this estimation is measured by moving the robot to the estimated pose by calculating the robot's pose using the image Jacobian. Fig. 12 shows the images obtained at the extimated pose of the top object at the moment of pick-up. The estimation error is calcualted as the difference between the estimated pose to pick and the objects' pose measured by the camera at the moment of pick-up. The estimation error is summarized in Table 2.

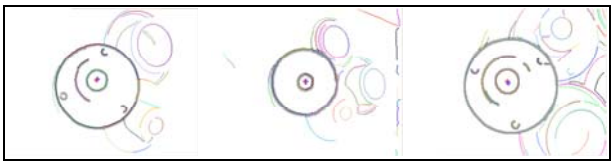


Fig. 12 Image of the top object at the moment of pick-up

Table 2. The estimated poses to pick and the measured poses by the camera at the moment of pick-up

Image number	Estimated by Kalman Filter	Measured Actually By Camera	Measurement Error
1	(148, 116, 1.3 , 1.2)	(151, 113, 1.2, 1.4)	(2, 3, 1, 0.2)
2	(157, 117, 1.5 , 1)	(157, 118, 1.5, 0.5)	(0, 1, 0, 0.6)
3	(158, 120, 3, 1.5)	(153, 117, 2.5, 2.2)	(5, 3, 0.5, 0.7)

The experimental results have shown that the pose estimation error is within the marginal error range so that the robot gripper can grasp the obejct without missing as long as the top object is successfully detected from the input image.

5 Conclusion

This paper proposed a solution for the top object picking problem to make the task be completed under the condition that the objects are arbitrarily stacked and moving on a conveyer belt. It consisted of two algorithms. One is to detect the top object from a stack of overlapped elliptical objects and to determine its pose in the image plane. The other one is to estimate the object's position to pick in the image plane using the Kalman estimator and to transform it to the world frame using the image Jacobian. From the experimental results, the followings have been confirmed: 1) the lighting system designed for reducing the effect of lights in the field enabled to collect a qualified input image from which the clear edges could be detected. 2) The ellipse detection algorithm using edge segments instead of edge pixels reduced the execution time by half and increased the accuracy significantly compared to the conventional RHT. 3) The estimation of object's position to pick was accurate enough so that the robot can pick up the object with a success rate of 98%. Although the proposed system has performed successfully with the elliptical objects, it is also confirmed that its speed problem, caused by still slow ellipse fitting process, should be solved to use in real-time environment.

References:

[1] K. Rahardja and A. Kosaka, "Vision-based Bin-Picking :Recognition and Localization of Multiple Complex Objects using Simple Visual Cues," IEEE/RSJ International Conference on Intelligent Robots and Systems, Osaka, Japan, November, 1996.

[2] Frank S.Cheng,Andrew Denman,"A Study of Using 2D Vision System for Enhanced Industrial Robot Intelligen ce",Proceedings of the IEEE International Conference on Mechatronics & Automation Niagara Falls,Canada, July 2005.

[3] Ren C. Luo, Woo Suk Yang and Yonghoon Kim,"Reco gnition of 3-D Objects using Modified 3-D General Hou gh Transformation".

[4] Robert A. McLaughlin," Technical Report – Randomi zed Hough Transform: Improved Ellipse Detection with Comparison"JP97-01, Univ of Western Australia, 1997.

[5] D.I. Kosmopoulos,C.S. Psomopoulos,"ROBUST ESTIMATION OF JACOBIAN MATRIX FOR IMAGE-BASED ROBOTIC VISUAL SERVOING TASKS", 14th IASTED, ICAMS, Benalmadena, Spain 2005.