PERGAMON Journal of the Franklin Institute 336 (1999) 237–255

Journal
of The
Franklin Institute

Multisensor-based robotic manipulation in an uncalibrated manufacturing workcell[1]

Bijoy K. Ghosh*, Di Xiao, Ning Xi, Tzyh Jong Tarn

Department of Systems Science and Mathematics, Washington University, Saint Louis, MO 63130, USA

Abstract

This paper deals with the problem of locating a part moving on a rotating disc conveyor with the aid of a single CCD camera and conveyor encoders and subsequently manipulating the part with a robot manipulator. The precise position and orientation of the disc conveyor with respect to the robot manipulator is assumed to be *a priori* unknown. Likewise the position and orientation of the camera with respect to the robot is also assumed to be *a priori* unknown. In order to manipulate the rotating part, the robot controller relies on data obtained from multiple sensors. Among the various sensors that have been used in this paper, we have a single CCD camera which is held fixed in the workspace of the robot and which is able to detect features of the part as it rotates with the conveyor. Likewise, the camera is also able to detect features of the end-effector as it moves in the workspace. Additionally, the conveyor is assumed to be equipped with an encoder that measures the angular position of the rotating disc. What makes the robotic manipulation problem hard is that the precise position of the camera and the robot are *a priori* assumed to be unknown. Thus, the path that the end effector needs to follow is not *a priori* known in the coordinates of the robot but needs to be computed during the course of the manipulation. In this paper, a new multisensor fusion scheme is proposed that performs this computation. The advantage of the scheme is that it greatly weakens requirement on the image processing speed. Overall, our algorithm is able to handle lack of calibration between the robot, the camera and the conveyor, advantage of which is that it significantly shortens the setup time of the manufacturing workcell. The paper touches upon both the theory and the experiments.
© 1998 The Franklin Institute. Published by Elsevier Science Ltd.

Keywords: Sensor fusion; Virtual rotation; Self calibration; Control

[1] Partially supported by the Department of Energy under grant number DE-FG02-90ER14140, by National Science Foundation under grant number CAD-9404949 and IRI-9706160, and by Sandia National Laboratories Contract No. AC 3752-C.

* Corresponding author. E-mail: ghosh@zach.wastl.edu

0016-0032/98/$ – See front matter © 1998 The Franklin Institute. Published by Elsevier Science Ltd.
PII: S0016-0032(98)00019-2

1. Introduction

Sensor-based control plays an important role in various flexible manufacturing problems. Utilizing sensors, a manufacturing system can compensate for changes in the environment and parameter uncertainties in its dynamic model. Vision sensors are useful in remote-sensing and vision-based robot control frequently extends the domain of application of a robot without an explicit need for intervention or reprogramming. Control of robot manipulators with vision in the feedback loop has an exciting history starting probably with the pioneering work of Hill and Park [1] and Weiss *et al.* [2]. Subsequent work in the area has focused on "Visual Servoing" wherein the emphasis is to visually locate the position and orientation of an object and to control a robot manipulator to grasp and manipulate the object. If the object is not stationary, then the process of locating the object and repositioning the robot through control must be repeated iteratively until the task has been accomplished. This leads naturally to real-time vision-based feedback and control problems that have been subsequently studied in [3, 4] and many others. As a result, many important tasks, such as Bolt Insertion, Conveyor Belt Picking, Weld Seal Tracking, Part Mating, Road Vehicle Guidance, Juggling, Fruit Picking, etc., to name a few, have been accomplished with the aid of computer vision. For a detailed discussion on "vision-based control of robot manipulators", we would like to refer to [5], a special issue of the IEEE Transactions on Robotics and Automation and to [6], a special issue of the Mathematical and Computer Modeling.

It may be emphasized however that one has to overcome many difficulties in order to utilize visual information. First of all, visual data is not always reliable. Vision systems could occasionally fail to generate any useful information but noise due to variation of illumination, overlapping of different workpieces or accidental obstruction of the camera. Video sensor imprecision as a real sensing device is an additional source of error. Secondly, image processing algorithms are always time-consuming. Therefore, use of visual information alone for the purpose of robot control could lead to poor accuracy or even instability. Furthermore, many of the earlier work in the area of visual servoing requires precise calibration data before control schemes are to be implemented. In this paper, the calibration problem is addressed by proposing a multisensor fusion based self-calibration scheme, which keeps the parameters updated as the manipulator is completing its task.

The multisensor fusion scheme is introduced for part manipulation in an uncalibrated manufacturing workcell. The workcell considered has been shown in Fig. 1 and consists of a rotating disc conveyor, an overhead CCD camera held fixed to the ceiling and a robot-manipulator on a mobile platform. The precise relationship between robot-mounted coordinate frame and the disc attached frame is assumed to be unknown. Likewise we also assume that the exact position and orientation of the camera on the ceiling is unknown. A new visually guided control and tracking algorithm has been proposed and experimentally demonstrated in this paper, for precise manipulation of a rotating part in a dynamic environment with imprecise and imprecise calibration parameters for both, the position/orientation of the manipulator base and the position/orientation of the camera.

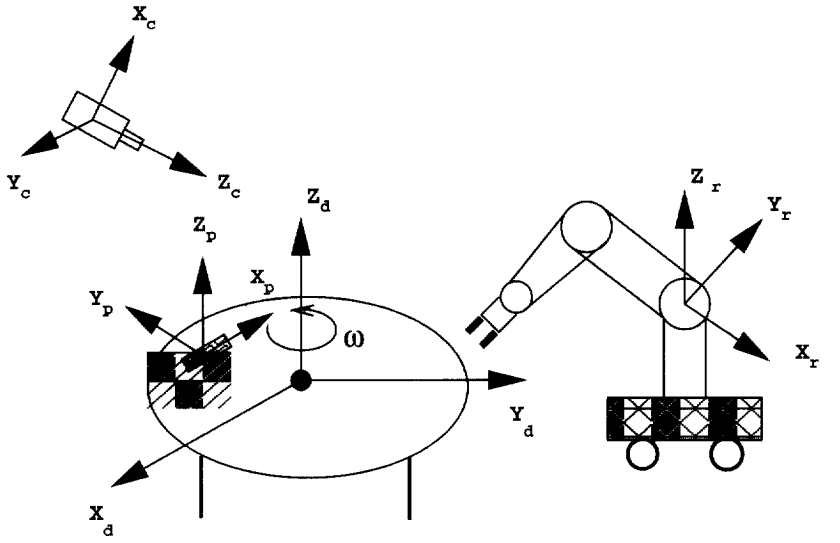

Fig. 1. A typical manufacturing workcell.

Prior work in this field includes research in the area of "Visual Servoing", see [2, 4]; "Vision-Based Parameter Estimation" see [7, 8]; and Multisensor Fusion, see [9, 10]. The process of multisensor fusion, as has been described in [10], is to combine information from various sensors to obtain inferences that may not be possible from single sensor alone. Multisensor fusion schemes, as have mostly been employed in the literature, increase the reliability of the observed sensor data by averaging over redundant sensory measurements. In this paper, however, we propose a multisensor fusion scheme that operates in a complementary rather than redundant mode. The purpose of our fusion process is to compensate for the relatively low speed of the image processing and to compensate for our ignorance about the calibration parameters.

This paper has been structured as follows. In Section 2, we present statements of the main problems considered and briefly describe the so-called "virtual rotation" algorithm already introduced in [11]. In Section 3, a new multisensor fusion scheme is introduced in order to compute the position, height and orientation of a three-dimensional part placed on a rotating disc conveyor. We also describe a feature point based calibration scheme for self-calibrating a robot coordinate frame in the workspace. In Section 4, we introduce sensor-based control problems in robotic manipulation. In Section 5, the experimental setup and detailed results of experiments on robotic manipulation are discussed. Finally in Section 6, the paper ends with a conclusion.

2. A virtual rotation algorithm in visually guided robotic manipulation

We consider a manufacturing workcell as shown in Fig. 1. The workcell is equipped with a rotating conveyor (equipped with encoders that measure the rotation angle), a robotic manipulator mounted on a mobile platform and a computer vision system with a single CCD camera. The task is to control the robot manipulator to pick up a part placed on the rotating conveyor. The key of the proposed sensor fusion scheme is to fuse the encoder readings with the observation of the camera to provide necessary information for part manipulation. We also make the following assumptions about our workcell.

A1. The precise position and orientation of the camera with respect to the robot coordinate frame is unknown. Additionally, the precise position and orientation of the conveyor with respect to the robot coordinate frame is also unknown.

A2. The plane of the conveyor and the XY-plane of the base frame of the robot are parallel.

A3. The part has known simple shape. In particular, we assume that observing feature points placed on the top surface of the part enables one to determine the orientation of the part.

A4. The entire workcell is in the view field of the camera. The center of the conveyor and a reference point on the conveyor is also assumed to be observed by the camera.

A5. The intrinsic parameters, viz., the focal length, etc., of the camera are known.

Usually, robots are controlled by first defining a task on an appropriately chosen task space. Defining a task in the three-dimensional task space consists of choosing a path that the robot needs to follow in order to pick the part from the conveyor. Since the part itself is moving with the conveyor, it is important to ascertain the motion of the part in the robot-coordinate frame before task planning is to be completed. In this section, we discuss the problem of estimating the position, orientation and the motion of the part. Note that, since the relative position of the camera and the robot with respect to the conveyor is not *a priori* known, part localization will also involve the self-calibration of different coordinate frames.

We do not assume that the camera has been selectively placed at any specific known position in the workcell. We now describe a virtual rotation algorithm to ascertain the relative position of the camera with respect to the coordinate frame of the disc conveyor. Note that if a camera looks straight down towards the conveyor (i.e. the image plane of the camera is parallel to the plane of conveyor), then the feature marks observed on the image plane are exactly, up to a fixed scale factor, the feature marks on the conveyor. Starting from an arbitrary initial position, the new position of the camera can be selected to be where its optical axis is perpendicular to the conveyor. The shape of the part is such that from this new position of the camera, the position and orientation of the part can be easily determined with respect to the coordinates of the conveyor.

In our actual implementation of the algorithm, we do not rotate the camera physically but instead process the observed image. Since every point on the conveyor undergoes a circular trajectory as the conveyor rotates , the image of the circle is an ellipse on the image plane. The shape of the ellipse depends on the relative orientation

of the camera with respect to the normal vector of the plane of the conveyor. In order to obtain the image from the top-view of the camera, we transform the observed image in such a way that the projected ellipse is transformed to a circle. The details of the steps are described as follows.

2.1. Virtual rotation algorithm

Let (X_r, Y_r) *be the coordinates of the reference point on the image plane. Let* (X_c, Y_c) *be the coordinates of the image of the center of the conveyor.*

Step 1. *A rotation matrix* R_k *of the camera around its optical center is applied to transform the image of the center of the conveyor to the center of the image plane.*

Step 2. *Ellipse parameters are estimated by observing the trajectory of the reference point on the image plane. Recursive least-squares fitting algorithm is used for this purpose.*

Step 3. *A rotation matrix* R_z *around Z-axis of the camera is applied to transform the major axis of the ellipse on the image plane into a position parallel to the Y-axis of the image plane. Note that this would automatically place the minor axis along X-axis.*

Step 4. *A rotation matrix* R_y *of the camera around the Y-axis of the image plane is applied to transform the ellipse to a circle. The rotation angle can be obtained in terms of the ellipse parameters determined in step* 2. *Note that this circle would automatically have its center on the X-axis of the image plane.*

The basic idea of rotation introduced in the above algorithm follows from similar rotation schemes introduced by Kanatani [12]. With the virtual rotation algorithm in hand, one can easily "virtually rotate" the original image data for further processing. Hereafter, we assume that all image data have been transformed by means of the above algorithm. Hence, without loss of generality, we may assume that the image plane of the camera is parallel to the plane of the conveyor.

In order to describe the position and orientation of the part (refer to Fig. 1 for details), several coordinate frames are defined. $O_dX_dY_dZ_d$ is the fixed disc conveyor frame with its origin being at the center of the conveyor and Z-axis being perpendicular to the conveyor. X-axis can be arbitrarily chosen on the disc conveyor. In our case, X-axis of the fixed disc conveyor frame is chosen to align with the initial reference line, which points towards the initial reference point from the disc-center. $O_bX_bY_bZ_b$ is the base frame of the robot and it moves with the platform of the robot. Note that the X and the Y axes are automatically selected in order that the projection of the center of the conveyor is exactly on the X-axis of the image plane. $O_aX_aY_aZ_a$ is the coordinate frame attached to the conveyor with its origin and Z-axis being the same as those of the fixed disc frame $O_dX_dY_dZ_d$ and X-axis pointing towards the reference point on the conveyor. The frame $O_aX_aY_aZ_a$ is assumed to rotate with the conveyor.

The position and orientation of a part on the conveyor is characterized by placing feature points on it. The coordinate ${}^bP(t)$ of any feature point on the part, with respect

to the base frame of the robot $O_bX_bY_bZ_b$ can be given by

$$^bP(t) = {}^bR_d\ {}^dR_a(\theta(t))^aP + {}^bT_d \tag{1}$$

where

$$^dR_a(\theta(t)) = \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 0 \\ \sin(\theta(t)) & \cos(\theta(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

^bR_d is the rotation matrix of $O_dX_dY_dZ_d$ with respect to $O_bX_bY_bZ_b$ and ^bT_d is the coordinate of the disc-center in the base frame $O_bX_bY_bZ_b$ of the robot; ^aP is the coordinate of the point with respect to the coordinate frame $O_aX_aY_aZ_a$ while $\theta(t)$ is the angle between X-axes of the fixed and attached disc frames $O_dX_dY_dZ_d$ and $O_aX_aY_aZ_a$.

Note that the relative position and orientation of the part with respect to the conveyor is time-invariant. It follows that, even though we estimate ^aP at a relatively low speed via the computer vision system, real-time trajectory of the part can still be updated at a high rate using encoders on the disc which measures the angle θ. This advantage is gained as a result of implementing multisensor fusion with complementary sensing information. Vision alone cannot update the geometry of the part in real time, whereas encoders working without the vision system cannot compute the relative orientation of the part.

In the next section, we shall show how a multisensor fusion scheme can be implemented to determine the relative position ^aP and the relation between the fixed disc frame and the base frame of the robot, ^bR_d and ^bT_d.

3. A self-calibration scheme for part estimation and robot calibration

The self-calibration scheme introduced in this section has the following two important purposes. First of all, it provides the position and orientation of a three-dimensional part on the conveyor with respect to the rotating coordinate frame $O_aX_aY_aZ_a$. Secondly, it provides us with an estimate of the coordinate transformation ^bR_d and ^bT_d between the coordinate frames $O_dX_dY_dZ_d$ and $O_bX_bY_bZ_b$. Since position and orientation of a part is completely specified using feature points, the first problem reduces completely to estimating ^aP, as has been introduced in Eq. (1).

The problem of estimating the coordinates ^aP in the rotating coordinate frame $O_aX_aY_aZ_a$ has already been introduced by Yu *et al.* [11], using the virtual rotation algorithm. Yu's work was however restricted to only planar parts on the conveyor. For a non-planar part, a feature point on the part is not necessarily on the plane of the conveyor, but at an unknown height h from it. Rotation algorithm itself is not enough to estimate the location of the feature point and a suitable modification is proposed in this section.

3.1. The problem of estimating the coordinates of a point with respect to the rotating coordinate frame

If $OXYZ$ is any coordinate frame with its origin O at the disc center and its Z-axis perpendicular to the conveyor, it is possible to compute the coordinates (x_p, y_p, h) of a point p in the three-dimensional space given the height h of the point p from the plane of the disc conveyor. The computation can be performed using parameters than can, in turn, be computed after virtual rotation of the camera. Recall that the virtual rotation is performed with respect to a reference line on the disc conveyor. In this section we shall assume that the reference line is chosen to be the line joining the disc center and a reference point with coordinates $(x_{\text{ref}}, y_{\text{ref}}, 0)$ with respect to $OXYZ$.

As it is apparent from Fig. 2, using simple geometry of projection, it can be shown that the coordinates x_p, y_p of a point p are computed as follows:

$$x_p = \frac{ax_{\text{ref}} - by_{\text{ref}}}{l^2} + \frac{(\bar{a} - a)x_{\text{ref}} - (\bar{b} - b)\ y_{\text{ref}}}{lLf} h \tag{3}$$

$$y_p = \frac{bx_{\text{ref}} + ay_{\text{ref}}}{l^2} + \frac{(\bar{b} - b)x_{\text{ref}} + (\bar{a} - a)x_{\text{ref}}}{lLf} h \tag{4}$$

where

$$a = \overrightarrow{C_oP_o} \cdot \overrightarrow{C_oR_o}; \quad b = \overrightarrow{C_oP_o} \times \overrightarrow{C_oR_o} \tag{5}$$

$$\bar{a} = \overrightarrow{C_oO_o} \cdot \overrightarrow{C_oR_o}; \quad \bar{b} = \overrightarrow{C_oO_o} \times \overrightarrow{C_oR_o} \tag{6}$$

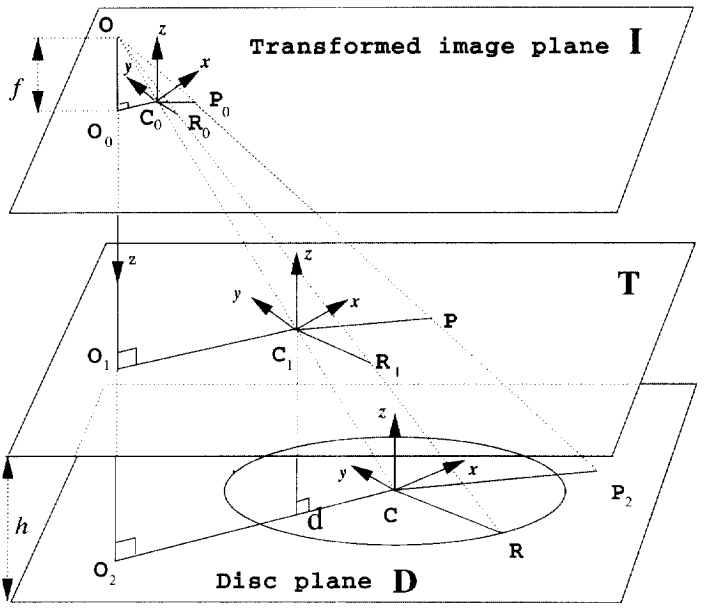

Fig. 2. Determination of the position of a point p.

in which $\overrightarrow{AB}$ denotes the vector from A to B while "$\cdot$" and "$\times$" are operators of dot product and cross product, respectively. The points C_o, R_o and P_o are the transformed images (after virtual rotation) of the disc center, the reference point and the point p, respectively. The point O_o is the intersection of the optical axis and the transformed image plane and h stands for the distance between the point p and the plane of the disc conveyor. The coordinates $(x_{ref}, y_{ref}, 0)$ are the coordinates of reference point in the frame $OXYZ$ and (x_p, y_p, h) the coordinates of the point p in the same frame. The scalar L denotes the length of the reference line segment while l is the length of the transformed image of the reference line segment and f is the focal length of the camera. Note that a, b, $\bar{a}$, $\bar{b}$ and l can be easily computed in terms of the image coordinates of C_o, R_o and P_o. From assumptions A4 and A5, about the part and the workcell, L and f are known constants. For the fixed camera, l is also a constant. Hence, if the coordinates (x_{ref}, y_{ref}) of the reference point are given in the frame $OXYZ$, the position of the point p in the same frame is an affine function of h, the distance of the point from the conveyor.

Equations (3) and (4) describe an affine line in the coordinate frame $OXYZ$. The affine line passes through the optical center of the camera and the feature point (x_p, y_p, h). Restricting the coordinate frame to $O_aX_aY_aZ_a$, consider a feature point on the part with coordinates (x_p, y_p, h) and suppose that two different images are taken at time t_1 and t_2. Applying the virtual rotation algorithm described in Section 2 to each of the two images, it follows that we can write

$$\begin{aligned} x_p &= \theta_1(t_i) + \theta_2(t_i)h \\ y_p &= \theta_3(t_i) + \theta_4(t_i)h \end{aligned} \tag{7}$$

for $i = 1, 2$. Note that Eq. (7) describes two different affine lines described by the feature point p at two different instants of time t_1 and t_2. The parameters θ_1, θ_2, θ_3, θ_4 can be correspondingly determined from Eqs. (3) and (4) and can take different values for different instances of time. However, since the part does not move in the rotating coordinate frame, it follows that (x_p, y_p, h) remain invariant at t_1 and t_2. From Eq. (7), we have

$$\begin{aligned} \theta_1(t_1) + \theta_2(t_1)h &= \theta_1(t_2) + \theta_2(t_2)h \\ \theta_3(t_1) + \theta_4(t_1)h &= \theta_3(t_2) + \theta_4(t_2)h \end{aligned} \tag{8}$$

which can be solved for h using least-squares method.

3.2. Estimation of coordinate transformation between disc conveyor and robot

In order to determine the relation between the fixed disc frame $O_dX_dY_dZ_d$ and the base frame $O_bX_bY_bZ_b$ of the robot, we need to describe a set of points in both frames since the two frames are related by

$$^bP = {}^bR_d\ {}^dP + {}^bT_d, \tag{9}$$

where ^bP and ^dP are the coordinates of a point in the base frame and the fixed disc frame, respectively.

Since the plane of the conveyor and the XY-plane of the base frame of the robot are parallel, it follows that there is only one unknown in the rotation matrix ${}^{b}R_{d}$. The vector ${}^{b}T_{d}$ has three unknown elements giving rise to a total of four parameters that are to be determined. In order to determine these parameters, one would need to know the coordinates ${}^{b}P$ and ${}^{d}P$ for at least two distinct points.

In this section, we shall choose the two points to be on the end-effector. The coordinates ${}^{b}P$ for each of the two points are obtained from the robot encoders. The coordinates ${}^{d}P$ for the two points need to be computed and is done as follows.

Let (x_1, y_1, z_1) and (x_2, y_2, z_2) be the coordinates of the two points on the end-effector w.r.t. the camera frame. On the image plane, the corresponding coordinates of the two points are (X_i, Y_i), $i = 1, 2$, where $X_i = fx_i/z_i$, $Y_i = fy_i/z_i$. Let s be the Euclidean distance between the two points and let us assume that $z_1 - z_2 = d$. It follows that we have

$$[(X_1 - X_2)^2 + (Y_1 - Y_2)^2]z_2^2 + 2[X_1(X_1 - X_2) + Y_1(Y_1 - Y_2)]\,dz_2 + (X_1^2 + Y_1^2 + f^2)d^2 - f^2s^2 = 0 \tag{10}$$

In general, the parameter "s" can be observed from the encoders on the robot. On the other hand, the parameter "d" cannot be computed unless the camera frame has been transformed using the virtual rotation algorithm. The difference between the Z-coordinates of the two points in the transformed camera frame has the same magnitude with opposite sign compared to the Z-coordinates of the two points in the coordinates of the base frame. The latter, of course, can be read from the robot encoders. In general, there are two solutions to Eq. (10). However, since Eq. (10) has always at least one real solution, it follows that Eq. (10) has two real solutions. In many cases, however, z_2 can be recovered uniquely. For instance, if $d = 0$ (i.e. the line segment joining the two points is parallel to the conveyor), then Eq. (10) reduces to

$$[(X_1 - X_2)^2 + (Y_1 - Y_2)^2]\,z_2^2 = f^2s^2 \tag{11}$$

indicating the existence of an unique positive solution for z_2. Defining,

$$F = [X_1(X_1 - X_2) + Y_1(Y_1 - Y_2)]d$$

$$E = (X_1 - X_2)^2 + (Y_1 - Y_2)^2$$

$$G = (X_1^2 + Y_1^2 + f^2)d^2 - f^2s^2$$

it follows that the two solutions of z_2 are

$$-\frac{F}{E}\left(1 \pm \sqrt{1 - \frac{EG}{F^2}}\right)$$

If $G < 0$, then the two solutions have opposite signs and therefore there is an unique $z_2 > 0$ which is the solution in front of the camera. Once z_2 is computed, x_2 and y_2 can be computed from z_2 using $x_2 = z_2X_2/f$, $Y_2 = z_2Y_2/f$. Recall that (x_2, y_2, z_2) are coordinates of the point in the transformed camera frame and hence z_2 is the vertical

distance between the point on the end effector and the optical center of the camera. If h_2 is the height of the point from the plane of the conveyor it follows that

$$h_2 = H - z_2 \tag{12}$$

where H is the height of the optical center given by $L/\ell f$. We conclude that

$$h_2 = \frac{L}{\ell} f + \frac{F}{E}\left(1 + \sqrt{1 - \frac{EG}{F^2}}\right)$$

where we already have ensured that $G < 0$. Once the height h_2 is known, the X and the Y coordinates can be computed using Eqs (3) and (4). This finishes our discussion on computing ${}^{\mathrm{d}}P$ for one of the two points on the end-effector. One computes ${}^{\mathrm{d}}P$ for the other point on the end-effector similarly.

Having obtained coordinates of two points on the end-effector with respect to $O_{\mathrm{d}}X_{\mathrm{d}}Y_{\mathrm{d}}Z_{\mathrm{d}}$ and $O_{\mathrm{b}}X_{\mathrm{b}}Y_{\mathrm{b}}Y_{\mathrm{b}}$, the rotation matrix ${}^{\mathrm{b}}R_{\mathrm{d}}$ and the translation vector ${}^{\mathrm{b}}T_{\mathrm{d}}$ can be easily computed by rewriting Eq. (9) as

$$\begin{bmatrix} {}^{\mathrm{b}}x_i \\ {}^{\mathrm{b}}y_i \\ {}^{\mathrm{b}}z_i \end{bmatrix} = {}^{\mathrm{b}}R_{\mathrm{d}} \begin{bmatrix} {}^{\mathrm{d}}x_i \\ {}^{\mathrm{d}}y_i \\ {}^{\mathrm{d}}z_i \end{bmatrix} + {}^{\mathrm{b}}T_{\mathrm{d}}, \quad i = 1, 2 \tag{13}$$

where $({}^{\mathrm{b}}x_i, {}^{\mathrm{b}}y_i, {}^{\mathrm{b}}z_i)$ and $({}^{\mathrm{d}}x_i, {}^{\mathrm{d}}y_i, {}^{\mathrm{d}}z_i)$ are the coordinates of the ith point with respect to the base frame and the fixed disc frame, respectively. Recall from Assumption A2, that ${}^{\mathrm{b}}R_{\mathrm{d}}$ has the following structure:

$${}^{\mathrm{b}}R_{\mathrm{d}} = \begin{bmatrix} r_{11} & r_{12} & 0 \\ -r_{12} & r_{11} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{14}$$

Eliminating ${}^{\mathrm{b}}T_{\mathrm{d}}$ from Eq. (13) we have

$$\begin{bmatrix} {}^{\mathrm{b}}e_1 \\ {}^{\mathrm{b}}e_2 \\ {}^{\mathrm{b}}e_3 \end{bmatrix} = {}^{\mathrm{b}}R_{\mathrm{d}} \begin{bmatrix} {}^{\mathrm{d}}e_1 \\ {}^{\mathrm{d}}e_2 \\ {}^{\mathrm{d}}e_3 \end{bmatrix}$$

where

$$\begin{bmatrix} {}^{\mathrm{b}}e_1 \\ {}^{\mathrm{b}}e_2 \\ {}^{\mathrm{b}}e_3 \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{b}}x_2 - {}^{\mathrm{b}}x_1 \\ {}^{\mathrm{b}}y_2 - {}^{\mathrm{b}}y_1 \\ {}^{\mathrm{b}}z_2 - {}^{\mathrm{b}}z_1 \end{bmatrix}, \quad \begin{bmatrix} {}^{\mathrm{d}}e_1 \\ {}^{\mathrm{d}}e_2 \\ {}^{\mathrm{d}}e_3 \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{d}}x_2 - {}^{\mathrm{d}}x_1 \\ {}^{\mathrm{d}}y_2 - {}^{\mathrm{d}}y_1 \\ {}^{\mathrm{d}}z_2 - {}^{\mathrm{d}}z_1 \end{bmatrix} \tag{15}$$

which are linear equations in r_{11} and r_{12}. As long as the line joining the two points is not parallel to Z-axis of the base frame, the equations always have a unique solution. However, such a solution may not satisfy the constraint $r_{11}^2 + r_{12}^2 = 1$ due to possible effect of noise in observed data and computation errors. In other words, ${}^{\mathrm{b}}R_{\mathrm{d}}$ obtained in this way may not be an orthogonal matrix.

The problem of determining ${}^{b}R_{d}$ can be viewed as an optimization problem of determining r_{11} and r_{12} with the structure in Eq. (14) such that

$$\left\| \begin{bmatrix} {}^{b}e_1 \\ {}^{b}e_2 \\ {}^{b}e_3 \end{bmatrix} - {}^{b}R_{d} \begin{bmatrix} {}^{d}e_1 \\ {}^{d}e_2 \\ {}^{d}e_3 \end{bmatrix} \right\|_2$$

is minimized subject to the constraint $r_{11}^2 + r_{12}^2 = 1$. Solving this optimization problem yields the solution

$$\begin{bmatrix} r_{11} \\ r_{12} \end{bmatrix} = \frac{1}{\sqrt{{}^{b}e_1^2 + e_2^2}\sqrt{{}^{d}e_1^2 + {}^{d}e^{22}}} \begin{bmatrix} {}^{b}e_1\,{}^{d}e_1 + {}^{b}e_2\,{}^{d}e_2 \\ {}^{b}e_1\,{}^{d}e_2 - {}^{b}e_2\,{}^{d}e_1 \end{bmatrix}$$

Having known ${}^{b}R_{d}$, we compute

$${}^{b}T_{d} = \begin{bmatrix} {}^{b}x_1 \\ {}^{b}x_2 \\ {}^{b}x_3 \end{bmatrix} - {}^{b}R_{d} \begin{bmatrix} {}^{d}x_1 \\ {}^{d}x_2 \\ {}^{d}x_3 \end{bmatrix}$$

4. Sensor-based control of a robotic manipulator

So far in this paper, we have seen that using a suitable fusion of the visual information with the measurements from the robot and disc encoders, one is able to compute unknown transformation parameters in Eq. (1). Since these parameters are time-invariant, the speed of the computation has not become an issue. What is required is that the rotation angle $\theta(t)$ is measured and updated in real time. In this paper, $\theta(t)$ is measured using an encoder on the turn table and is therefore updated in real time.

The position, orientation and velocity of the part in the base frame of the robot can be computed using Eq. (1). We now describe how a robot can be controlled to pick a moving part on the rotating disc conveyor.

The dynamic model of a six-degrees-of-freedom robot arm is given by

$$\tau = D(q)\ddot{q} + C(q, \dot{q}) + G(q) \tag{16}$$

and the position and orientation output of the end effector in the task space is given by

$$Y = h(q) = [h_1(q)\ h_2(q)\ h_3(q)\ h_4(q)\ h_5(q)\ h_6(q)]^{T} \tag{17}$$

where $Y = [x\ y\ z\ O\ A\ T]^{T}$, q is the joint angle vector, $D(q)$ the inertia matrix, $C(q, \dot{q})$ the centripetal and Coriolis terms, $G(q)$ the gravity loading and τ the joint torque vector. Note that the vectors $q, \tau \in \mathbb{R}^6$. For details about $O\ A\ T$ representation of the orientation, we would like to refer to [13].

The robot dynamic model, (16) and (17), are nonlinear equations. It is very difficult to directly design a tracking control law for such a nonlinear dynamics. Instead, the nonlinear feedback technique [14] used to linearize and decouple the above dynamic model in the task space and bring the nonlinear control problem to a linear control problem.

Let $x_1 = q$, $x_2 = \dot{q}$ and $E(x_1, x_2) = C(x_1, x_2) + G(x_1)$. Equation (16) can be rewritten in a standard nonlinear state-space form

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -D^{-1}(x_1)E(x) \end{bmatrix} + \begin{bmatrix} 0 \\ D^{-1}(x_1) \end{bmatrix} \tau$$

Therefore, the robot dynamic model could be stated as

$$\begin{aligned} \dot{x} &= f(x) + g(x)\tau \\ Y &= h(x_1) \end{aligned} \tag{18}$$

where $x = [x_1, x_2]^T$. Using results of Differential Geometric Control Theory [14], there exists a diffeomorphic state transformation $T(x)$ and a nonlinear feedback law $\tau = \alpha(x) + \beta(x)v$ which linearizes and decouples the robot dynamics. The diffeomorphic state transformation $T(x)$ is given by

$$z = T(x) = [h_1(x_1), L_f h_1(x_1), \ldots, h_6(x_1), L_f h_6(x_1)]^T,$$

and the nonlinear feedback law is

$$\tau = \alpha(x) + \beta(x)v$$

with

$$\alpha(x) = -D(x_1)J_h^{-1} \begin{bmatrix} L_f^2 h_1(x_1) \\ \vdots \\ L_f^2 h_6(x_1) \end{bmatrix}$$

$$= -D(x_1)J_h^{-1}[\dot{J}_h\dot{q} - J_h D^{-1}(x_1)E(x)] \tag{19}$$

$$\beta(x) = D(x_1)J_h^{-1} \tag{20}$$

where h_i is the ith component of $h(q)$ and L_f^k denotes the kth Lie derivative of $h(x)$ along the vector field $f(x)$, and J_h is the output Jacobian matrix of $h(x_1)$. In the transformed state z with the auxiliary input v, Eq. (18) appears in the Brunovsky canonical form as follows:

$$\dot{z} = Az + Bv \tag{21}$$

$$Y = Cz \tag{22}$$

Here A, B, C are block diagonal matrices. To see the structure of the above equations we write them in a more detailed fashion. Equations (21) and (22) represent six linear

and decoupled subsystems in the form

$$\dot{z}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} z_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v_i, \quad y_i = [1 \quad 0]\, z_i$$

where $z_i = [h_i \; L_f h_i]^T$. Each identical subsystem has double poles at the origin, therefore the system is not asymptotically stable. Introducing the feedback law

$$v_i^* = v_i - F_i z_i, \quad i = 1, \ldots, 6$$

where $F_i = [f_{i_1} f_{i_2}]$, the final form of the closed-loop system is described as follows:

$$\dot{z}_i = \begin{bmatrix} 0 & 1 \\ -f_{i_1} & -f_{i_2} \end{bmatrix} z_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v_i^*, \quad y_i = [1 \quad 0]\, z_i$$

Note that F_i represents a linear PD controller. Therefore, the nonlinear feedback control law is given by

$$\tau = D(q) J_h^{-1} [\ddot{Y}^d(t) + K_v \dot{e}(t) + K_p e(t) - \dot{J}_h \dot{q}] + C(q, \dot{q}) + G(q) \tag{23}$$

where

$$e(t) = Y^d(t) - Y(t), \quad \dot{e}(t) = \dot{Y}^d(t) - \dot{Y}(t).$$

and $Y^d(t)$ and $\dot{Y}^d(t)$ are desired position and velocity of location and orientation of the part on the rotating disc conveyor. The system structure can be illustrated by Fig. 3. In Fig. 3, the sensing system generates the desired commands Y^d, $\dot{Y}^d$ and $\ddot{Y}^d$ based on the dynamics of the part and position of the robot. The linear controller ensures stability and transient response of the robot system. The nonlinear feedback actually linearizes and decouples the robot dynamic model. As a result, the robot systems can directly respond to the task level commands generated by the sensing system.

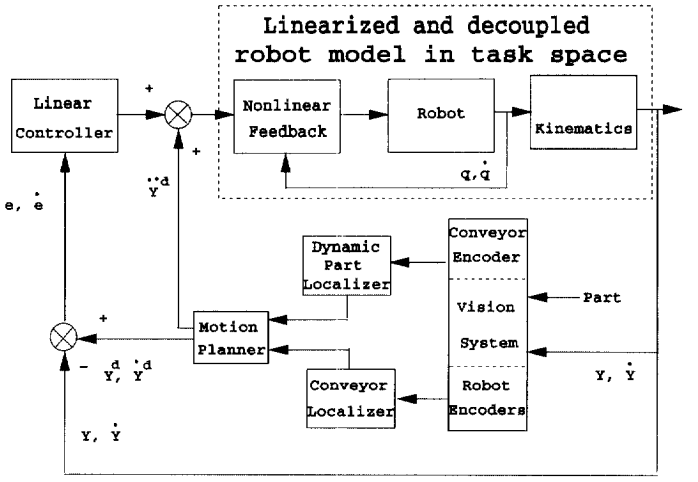

Fig. 3. Block diagram for planning and control of the robot.

5. Experimental setup and results

An experimental system has been set up in the Center for Robotics and Automation at Washington University. It consists of one PUMA 560 manipulator and a disc conveyor with the diameter of 0.9 m. The computer vision system consists of a CCD camera with image resolution of 256×256 and the Intelledex Vision processor based on a 16 MHz Intel 80386 CPU. It interfaces to the host computer, a SGI 4D/340 VGX. Visual measurements are sent to SGI by a parallel interface. The robot is controlled by UMC controller that also interfaces to SGI through memory mapping.

As described in earlier sections, this paper focuses on a multisensor fusion scheme for self-calibration of a robot manipulator and for localization of a non-planar part rotating on a disc conveyor. The experiments described here do not include tracking a rotating part by the robot. A similar tracking scheme has been tested experimentally for the case where the calibration between the robot and the disc conveyor is precisely known and the part is assumed to be planar [11]. Multisensor fusion and self-calibration schemes leading upto the estimation of the pose of a part on the conveyor with respect to the base frame of the robot has been detailed here. In all the experiments described in this section, the precise poses of the camera, the robot and the disc conveyor are assumed to be unknown. We have placed a part on the disc conveyor and then rotated it 180° such that the part would be in the workpace of the robot. When the disc conveyor stops, the pose of the part is computed. The end-effector of the robot is automatically controlled to approach the part, pick it up and move to a pre-specified location.

5.1. Results of experiment 1

In this experiment, we assume that the distance between the end-effector of the robot and the disc conveyor is unknown. Additionally, we assume that the height of the part is unknown (the actual height of the part is 4 cm).

Figure 4 shows the estimated and actual pose of the part. Figure 4a–c indicates the estimated and actual trajectory of the part in X-, Y- and Z-directions of the base frame of the robot, respectively. Figure 4d shows the estimated and actual orientation of the part in the base frame of the robot. In Fig. 4, the dotted lines denote the actual values and the solid lines are the estimated ones. The estimation errors of the pose of the part are illustrated in Fig. 5, where Fig. 5a–c show the position errors in X-, Y- and Z-directions, respectively. The error in orientation is provided in Fig. 5d. The position errors in X- and Y-directions are around 1 cm and the orientation error is less than 5°.

5.2. Results of experiment 2

In this experiment, we assume that the distance between the end-effector of the robot and the disc conveyor is known. Actually, we assume that the height of the disc conveyor in the base frame of the robot is known. As in experiment 1, the height of the part is assumed to be unknown. We carried out the experiment for two different heights of the parts. The actual heights of the parts are 4 and 7 cm, respectively.

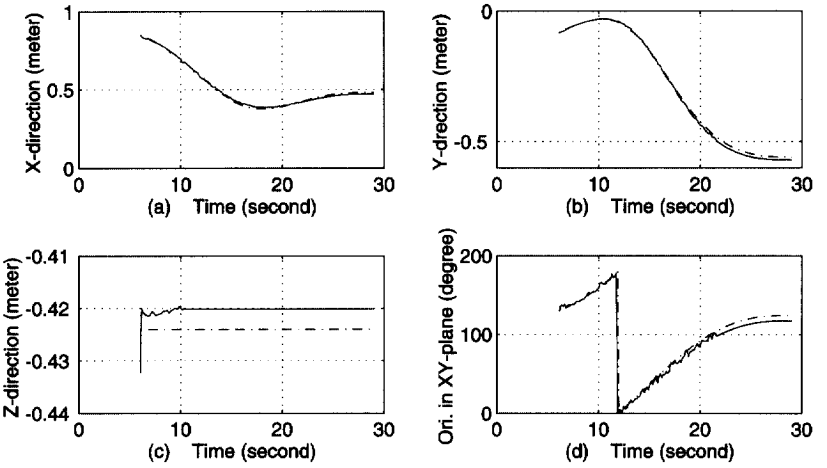

Fig. 4. Estimated and actual pose of the part for the case where the distance of the end-effector from the disc conveyor is unknown.

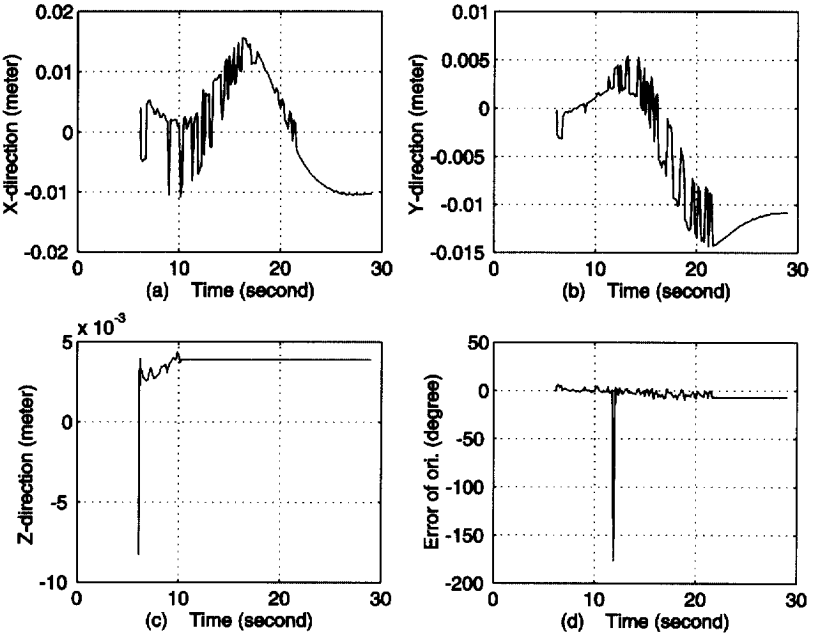

Fig. 5. Errors in pose estimation for the case where the distance of the end-effector from the disc conveyor is unknown.

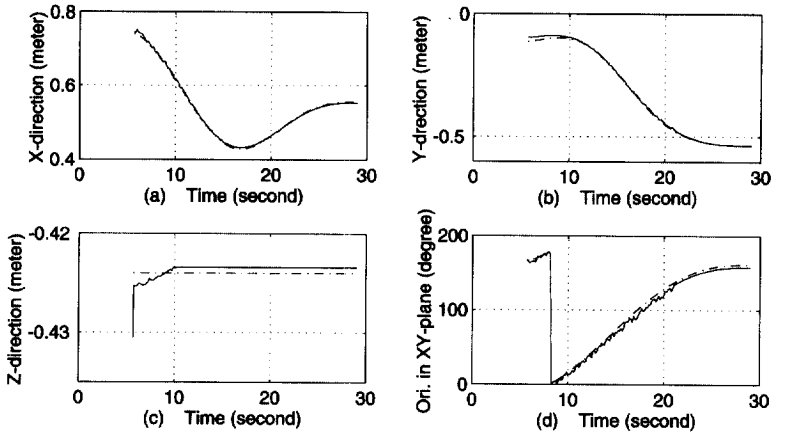

Fig. 6. Estimated and actual pose of the part for the case where the distance of the end-effector from the disc conveyor is known and height of the part is 4 cm.

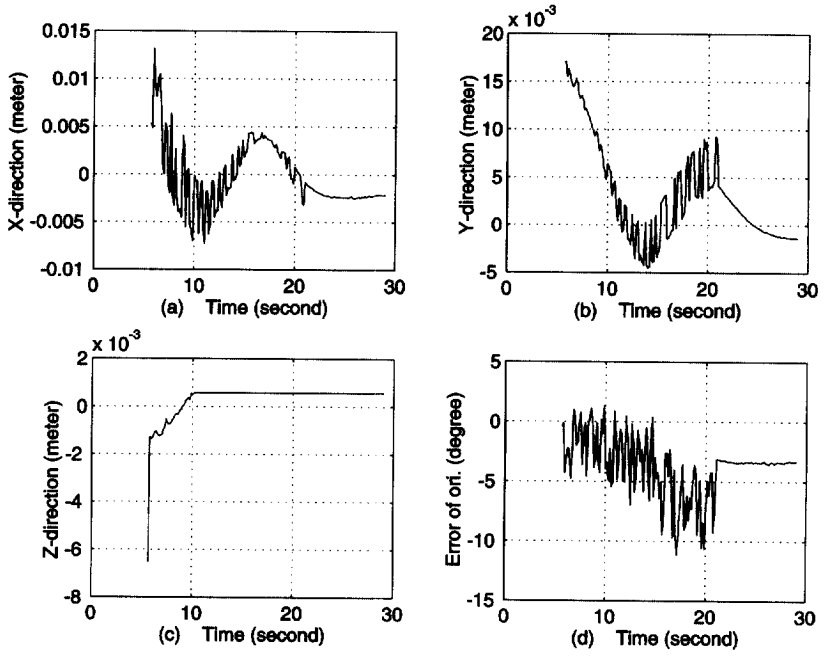

Fig. 7. Errors in pose estimation for the case where the distance of the end-effector from the disc conveyor is known and height of the part is 4 cm.

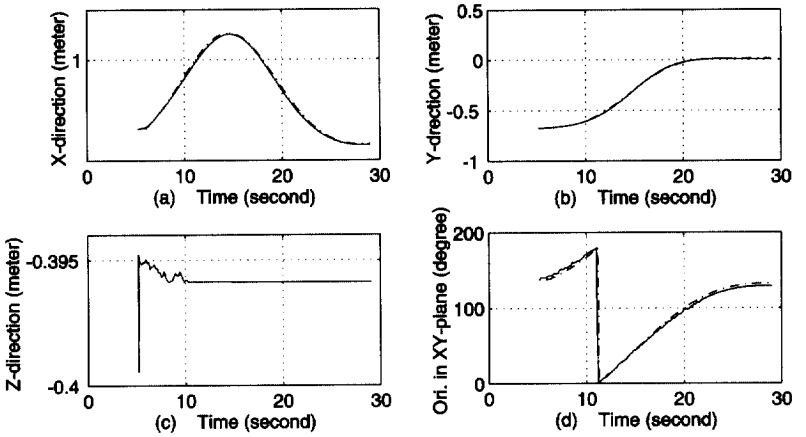

Fig. 8. Estimated and actual pose of the part for the case where the distance of the end-effector from the disc conveyor is known and height of the part is 7 cm.

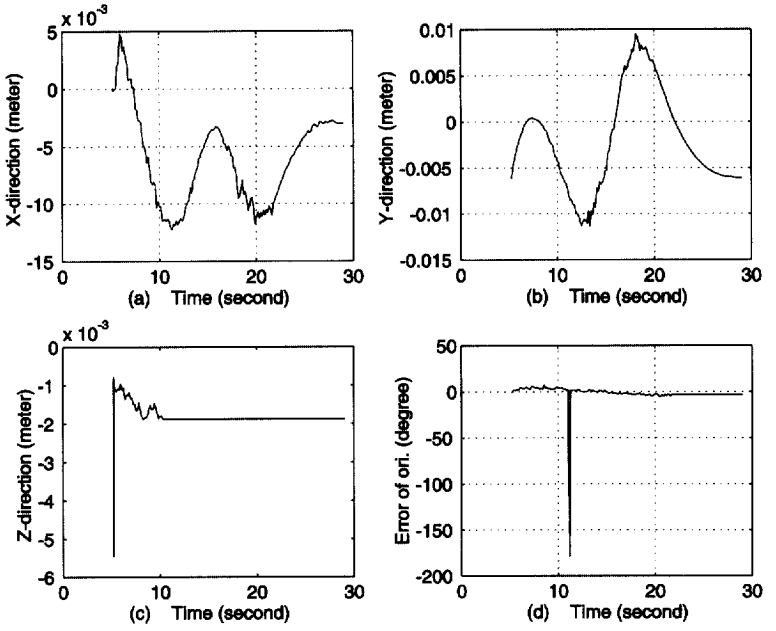

Fig. 9. Errors in pose estimation for the case where the distance of the end-effector form the disc conveyor is known and height of the part is 7 cm.

The experimental results for the case where the height of the part is 4 cm are shown in Figs. 6 and 7. The estimated and actual pose of the part are presented in Fig. 6, where a–c depict the estimated and actual trajectory of the part in X-, Y- and Z-directions of the base frame of the robot. In the figure, the estimated trajectory of

the part is denoted by the solid lines while the actual trajectory of the part is illustrated by the dotted lines. Figure 7 shows the estimation errors of the pose of the part. Figure 7a–c present the error in X-direction, Y-direction and Z-direction, respectively. Estimated and actual orientation is given in Fig. 6d whereas the error in orientation is given in Fig. 7d.

The experimental results for the case where the height of the part is 7 cm are illustrated likewise in Figs. 8 and 9. As is evident from the figures of this experiment, the position errors are within 1 cm and the orientation errors are less than 5°.

In comparing the results of experiments 1 and 2, the errors in position and orientation are somewhat comparable. In repeated trials, it is observed that the success rate (in terms of successfully grasping the part) in experiment 1 is lower than that of experiment 2. A quantitative comparison has not been made.

6. Conclusion

In this paper, we have introduced a multisensor-based self-calibration scheme for estimating the position and orientation of a part on a rotating conveyor with the eventual goal of manipulating the part with a robot manipulator, precise position of which is assumed to be unknown. The scheme we introduce computes the transformation parameters between the base frame of the robot and the coordinate frame on the conveyor. We also compute the height of the feature points on the part, improving an earlier work by Yu *et al.* [11], where the height has been assumed to be negligible.

References

[1] J. Hill, W.T. Park, Real time control of a robot with a mobile camera, Proc. 9th ISIR, Washington DC, March 1979, pp. 233–246.

[2] L. Weiss, A. Sanderson, C. Neuman, Dynamic sensor based control of robots with visual feedback, IEEE J. Robotics Automat. RA-3 (5) (1987) 404–417.

[3] B.H. Yoshimi, P.K. Allen, Alignment using an uncalibrated camera system, IEEE J. Robotics Automat. RA11 (4) (1995) 516–521.

[4] P.I. Corke, Visual control of robot manipulators—a review, in: K. Hashimoto (Ed.), Visual Servoing, World Scientific, Singapore, 1994, pp. 1–32.

[5] G.D. Hager, S. Hutchinson (Eds.), Special section on vision-based control of robot manipulators, IEEE Trans. Robotics Automat. RA-12 (5) (1996).

[6] B.K. Ghosh, N. Papanikolopoulos (Eds.), Modeling Issues in Visual Sensing, Math. Computer Modeling 24 (5, 6) (1996).

[7] T.S. Huang, A.N. Netravali, Motion and structure from feature correspondences—a review, Proc. IEEE 82 (2) (1994) 252–268.

[8] Y. Wu, S. S. Iyengar, R. Jain, S. Bose, A new generalized computational framework for finding object orientation using perspective trihedral angle constraint, IEEE Trans. PAMI 16 (0) (1994) 961–975.

[9] T.C. Henderson, E. Shilcrat, Logical sensor systems, J. Robotic Systems 1 (2) (1984).

[10] D.L. Hall, Mathematical Techniques in Multisensor Data Fusion, Artech House, Boston, London, 1992.

[11] Z. Yu, B.K. Ghosh, N. Xi, T.J. Tarn, Multi-sensor based planning and control for robotics manufacturing systems, Proc. Int. Conf. on Intelligent Robotics and Systems, August, vol. 3 (1995) 222–227.

[12] K. Kanatani, Group Theoretical Methods in Image Understanding, Springer, Berlin, 1990.
[13] T.J. Tarn, A.K. Bejczy, G.T. Marth, A.K. Ramadorai, Kinematic characterization of the PUMA 560 manipulator, Robotics Laboratory Report SSM-RL-91-15, Dept. of Systems Science and Mathematics, Washington University, St. Louis, USA, December 1991.
[14] A. Isidori, Nonlinear Control Systems, 2nd ed., Springer, Berlin, 1989.
[15] A.J. Koivo, N. Houshangi, Real-time vision feedback for servoing robotic manipulator with self-tuning controller, IEEE Trans. Systems Man Cybernet. 21 (1) (1991) 134–142.
[16] E.D. Dickmanns, V. Graefe, Dynamic monocular machine vision, Machine Vision Appl. 1 (1988) 223–240.
[17] J.T. Feddema, O.R. Mitchell, Vision-guided servoing with feature-based trajectory generation, IEEE J. Robotics Automat. RA-5 (5) (1989) 691–700.
[18] N. Papanikolopoulos, P.K. Khosla, T. Kanade, Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision, IEEE J. Robotics Automat. RA-9 (1) (1993) 14–35.
[19] R.I. Hartley, Estimation of relative camera positions for uncalibrated cameras, Proc. 2nd E.C.C.V. Santa Margherita Ligure, Italy, May 1992, pp. 579–587.
[20] M.A. Fischler, R.C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Comm. ACM 24 (6) (1981) 381–355.
[21] T.C. Henderson, W.S. Fai, C. Hansen, MKS: a multisensor kernel system, IEEE Trans. Systems, Man Cybernet. SMC-14 (5) (1984).
[22] H.F. Durrant-Whyte, Integrating distributed sensor information, an application to a robot system coordinator, Proc. IEEE Int. Conf. on Systems, Man Cybernet. (1985) 415–419.
[23] H.F. Durrant-Whyte, Consistent integration and propagation of disparate sensor observations, Proc. IEEE Int. Conf. on Robotics and Automation. San Francisco, CA, April 1986.
[24] R.C. Luo, M. Kay, Multisensor integration and fusion, SPIE 1988 Conf. on Multisensor Fusion, Orendo, FL, April 1988.
[25] R.C. Luo, M.H. Lin, R.S. Scherp, The issues and approaches of a robot multi-sensor integration, Proc. IEEE Int. Conf. on Robotics and Automation. Raleigh, NC, 1987, pp. 1941–1946.
[26] R.C. Luo, M.H. Lin, Intelligent robot multi-sensor data fusion for flexible manufacturing systems, in: Advances in Manufacturing Systems Integration and Processes, Proc. 15th Conf. on Production Research and Technology, U.C. Berkeley, Berkeley, CA, January 1989.
[27] P. Allen, R. Bajcsy, Object recognition using vision and touching, Proc. 9th Joint Conf. on Artificial Intelligence, Los Angeles, August 1985.
[28] A. Flynn, Redundant sensors for mobile robot navigation, MS Thesis, Dept. of EECS, MIT, Cambridge, MA, July 1985.
[29] M.J. Magee, B.A. Boyter, C.H. Chien, J.K. Aggarwal, Experiments in intensity guided range sensing recognition of three-dimensional objects, IEEE Trans. PAMI, PAMI-7 (6) (1985) 629–637.
[30] N. Nandhakumar, J.K. Aggarwal, Synergetic analysis of thermal and visual images for scene perception, Proc. Platinum Jubilee Conf. on Systems and Signal Processing, Indian Institute of Science, Bangalore, India, December 1986.
[31] A. Mitiche, J.K. Aggarwal, Multiple sensor integration/fusion through image processing—a review, Opt. Engng. 25 (3) (1996) 380–386.