Proceedings of the 2001 IEEE
International Conference on Robotics & Automation
Seoul, Korea - May 21-26, 2001

SmartView: Hand-Eye Robotic Calibration for Active Viewpoint Generation and Object Grasping

Yuichi Motai and Akio Kosaka
School of Electrical and Computer Engineering
Purdue University, West Lafayette, IN 47907, U.S.A.

Abstract

Viewpoint calibration is a method to manipulate hand-eye for generating calibration parameters for active viewpoint control and object grasping. In robot vision applications, accurate vision sensor calibration and robust vision-based robot motion control are essential for developing an intelligent and autonomous robotic system. This paper presents a new approach to hand-eye robotic calibration for vision-based object modeling and grasping. Our method provides a 1.0-pixel level of image registration accuracy when a standard Puma robot generates an arbitrary viewpoint. In order to attain this accuracy, our new formalism of hand-eye calibration deals with a lens distortion model of a vision sensor and utilizes a new parameter estimation algorithm using an extended Kalman filter. We demonstrate the power of this new method called "SmartView" for (1) generating 3D object models using an interactive 3D modeling editor, (2) recognizing 3D objects using stereo vision systems, and (3) grasping 3D objects using a manipulator. Experimental results using a Puma robot are shown.

1 Introduction

We will report an improved technique for calibrating wrist-mounted robotic vision systems. Our approach allows the system to compute robotic calibration parameters of both the camera and the end-effector for active viewpoint generation, after the system performs the vision-based camera calibration for only a small number of viewpoints.

Ordinarily, if one wishes to integrate multiple views for modeling 3D objects, one would carry out camera calibration for each viewpoint separately. An alternative consists of carrying out calibration for a certain number of designated viewpoints and using interpolation for other viewpoints. The disadvantages of this approach are (1) overall low accuracy and sometimes partially dominant error, and (2) spatial limitation of viewpoint generation. In our proposed scheme, we use only five viewpoints for calibration of the camera mounted on the gripper, and optimally estimate all necessary parameters for active control of viewpoints and precise object grasping. The advantages of our method are to minimize the calibration error by (1) applying a lens distortion model and (2) optimizing the camera parameters with robotic arm kinematics.

Since camera calibration is fundamental to all phases of research described in robot vision, much work has been done so far. These assumed a pin-hole model for the camera, an assumption that in early days was often a poor approximation to actual camera lenses, but is applicable today only if the optical quality of the camera lenses has improved. While fews of these references deal with non-linear aspects of camera imaging [10], others address the issue of efficient calculation of the various camera parameters [11]. Most papers on hand-eye calibration ignore the lens distortion (since it is difficult to model), and camera intrinsic parameters are chosen just from one of the viewpoints. In our work, in order to achieve an accurate and reliable performance of active vision control, we have modeled a camera with lens distortion as well as have optimally estimated camera calibration parameters by integrating those estimates obtained from multiple viewpoints.

In order for readers to get the sense of the power of our hand-eye calibration method, we show in Fig. 1 the image registration error of the overall hand-eye calibration for viewpoint generation. In this figure, small black circles are used for the camera calibration pat-

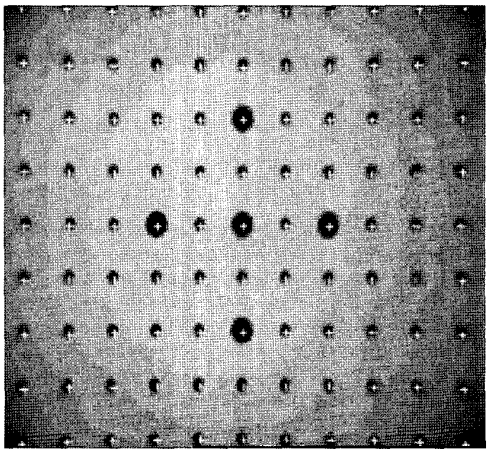

Figure 1: *The camera mounted on the gripper takes an image of calibration patterns of radius* 2.0 *mm at the distance of* 0.2 *m. White cross-bars are superimposed at the expected pattern centers using the estimated calibration parameters. This figure therefore demonstrates the accuracy of our calibration approach.*

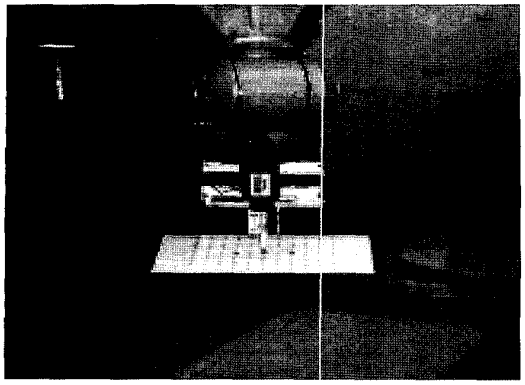

Figure 2: *Verification of hand-eye calibration using a stylus is shown. The robot attempts to locate the tip of the stylus to the origin of the world coordinate frame by using the estimated calibration parameters.*

terns in order to deal with lens distortion, and white cross-bars are superimposed to verify the accuracy of the method. As shown in this figure, we attain the 1.0-pixel level of image registration accuracy (1 *mm*-level of accuracy in the 3D workspace) when an arbitrary viewpoint is commanded to achieve. Fig. 2 also shows the verification of our hand-eye calibration for object grasping. The robot attempts to locate the tip of the stylus mounted on the gripper to the center of calibration patterns.

As several early studies formalized, the solution of the kinematics can be derived as a closed-form by inverse transformation matrix [4]. In order to determine the position and orientation of hand-eye camera with respect to the workspace, more recent issues are to solve the unknown transformation of the robot hand, which is formed as a homogeneous matrix equation $AX = XB$ [8, 2] and to solve this equation using a quaternion approach [3, 2]. In our formalism, we propose a modified algorithm using a non-linear iteration method and then evaluate the new solution by our current robot manipulation systems.

Our robotic calibration can be applied to develop a human-assisted model acquisition system [7]. In this project, objects are placed in the work area by a human who then "guides" the system into establishing image-to-image and pose-to-pose correspondences. After a model is acquired in this manner during the test phase, the same object in a random pose is viewed from two viewpoints for pose calculation. If the object is placed in the appropriate field, or randomly placed in the field of the single camera mounded on the robot hand, the robot hand can pick up the object. The computer vision checks which surface has grasping capabilities before picking up the object. In the acquired model, the object surfaces are classified as appropriate to be picked up or not, as shown in Fig. 3. And finally, robot manipulation is carried out to grasp the object in which the grasping points are already acquired.

We also seek a robust imaging so that our robot vision system can compute the reliable object pose. The imaging viewpoint decision has been studied as active vision. These manipulate a camera to improve the quality of the perceptual results. Because we can compute calibration parameters at any camera position, the active control of viewpoint direction increases machine perception activity, or the sensor planning [9].

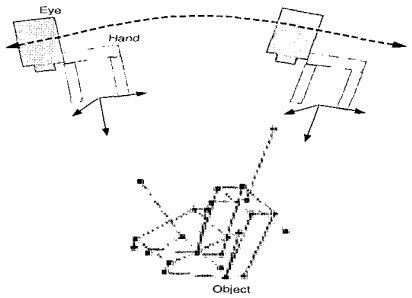

Figure 3: *Grasping coordinates in the object model.*

In this paper, we will first present our new approach of hand-eye calibration system and will then apply this calibration result to the human-assisted model acquisition system. Finally, experimental results which verify the power of our approach will be shown.

2 Problem Statement of Hand-Eye Calibration

We mount a monocular camera on the gripper of the robotic manipulator to generate a 3D object model by taking multiple views of the object from different viewpoints. As we mentioned, we like the robot to automatically generate all multiple views and compute all necessary calibration parameters for model acquisition systems. In such a strategy for 3D modeling, the hand-eye calibration is an important task to achieve. In order to formulate the hand-eye calibration problem, we define the following coordinate frames as shown in Fig. 4.

Base coordinate frame : $Base(x_B, y_B, z_B)$
This frame is associated with the robot base.

Tool coordinate frame : $Tool(x_T, y_T, z_T)$
The gripper of the robot is mounted with respect to this tool coordinate frame. The homogeneous transformation from the robot base to the tool is fully determined by the robot positioning commands.

Object coordinate frame : $Object(x_O, y_O, z_O)$
The object model is generated with respect to this coordinate frame.

World coordinate frame : $World(x_W, y_W, z_W)$
The world coordinate frame is specified in the workspace of the robot gripper.

Camera coordinate frame : $Camera(x_C, y_C, z_C)$
This coordinate frame is associated with the camera mounted on the gripper.

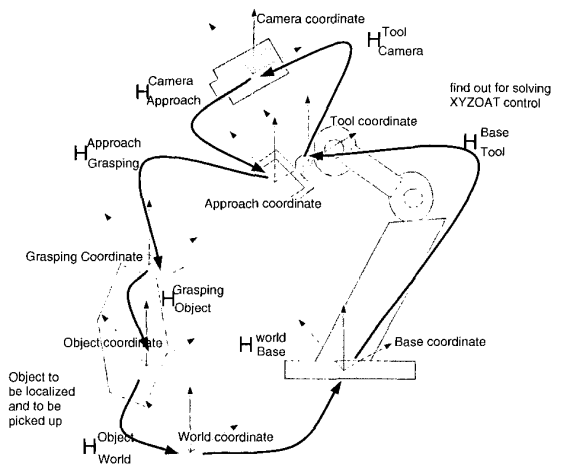

Figure 4: *Coordinate transitional illustration for robot manipulation.*

Image coordinate frame : *Image*(u, v)
The image coordinate frame is associated with the 2D camera image plane of the camera.

Since a Puma hand motion is fully determined by its positioning commands $XYZOAT$, we will be able to determine the homogeneous transformation $\boldsymbol{H}_{Base}^{Tool}$ from the robot tool to the robot base coordinate frames (See later Eqs. (18) and (19)). So our hand-eye calibration problem is described as follows:

- Determine the homogeneous transformations of $\boldsymbol{H}_{Camera}^{Tool}$ and $\boldsymbol{H}_{Base}^{World}$, as well as the camera-image transformation $\boldsymbol{g}$, namely,

$$\begin{bmatrix} u \\ v \end{bmatrix} = \boldsymbol{g}(x_C, y_C, z_C) \tag{1}$$

2.1 Camera image transformation

Before discussing the details of our calibration procedure, we will present the camera image formation which deals with lens distortion. Let us first assume that the camera is modeled by a pinhole camera. In this case, the camera image coordinates (u, v) can be normalized into the normalized pinhole camera image coordinates (u', v') by the intrinsic camera parameters – magnification factors α_u, α_v and the image center coordinates (u_0, v_0) as follows:

$$u' = \frac{u - u_0}{\alpha_u} = \frac{x_C}{z_C} \quad v' = \frac{v - v_0}{\alpha_v} = \frac{y_C}{z_C}. \tag{2}$$

If we deal with the lens distortion, we add the deviation to the normalized pinhole camera image coordinates (u', v'). Let $(\tilde{u}, \tilde{v})$ be the normalized camera image coordinate frame with lens distortion, then we can represent the relationship between (u', v') and $(\tilde{u}, \tilde{v})$ by the following equations [11]:

$$\begin{aligned} u' &= \frac{x_C}{z_C} \approx \tilde{u} + k_1 \tilde{u} \left(\tilde{u}^2 + \tilde{v}^2\right) \\ v' &= \frac{y_C}{z_C} \approx \tilde{v} + k_1 \tilde{v} \left(\tilde{u}^2 + \tilde{v}^2\right) \end{aligned} \tag{3}$$

where the coefficient k_1 for the deviation represents the parameter for the radial distortion of the camera, and $(\tilde{u}, \tilde{v})$ is computed from the actual camera image coordinates (u, v) as

$$\tilde{u} = \frac{u - u_0}{\alpha_u} \quad \tilde{v} = \frac{v - v_0}{\alpha_v}. \tag{4}$$

In this paper, we will call $\boldsymbol{s} = [\alpha_u, \alpha_v, u_0, v_0, k_1]^T$ the *intrinsic camera parameters*. Note that these intrinsic camera parameters will remain constant over the hand-eye calibration.

We now consider multiple viewpoints by moving a camera in the world coordinate frame. At viewpoint i, the camera location in the world coordinate frame is specified by the homogeneous transformation $\boldsymbol{H}_{Camera}^{World}$ which consists of the rotation matrix $\boldsymbol{R}^i = (r_{km}^i)$ and the translation vector $\boldsymbol{t}^i = [t_x^i, t_y^i, t_z^i]^T$. Then from Eqs. (3), 3D point (x_W, y_W, z_W) in the world coordinate frame is mapped onto the camera image frame as

$$u' = \frac{r_{11}^i x_W + r_{12}^i y_W + r_{13}^i z_W + t_x^i}{r_{31}^i x_W + r_{32}^i y_W + r_{33}^i z_W + t_x^i} \approx \tilde{u} + k_1 \tilde{u} \left(\tilde{u}^2 + \tilde{v}^2\right) \tag{5}$$

$$v' = \frac{r_{21}^i x_W + r_{22}^i y_W + r_{23}^i z_W + t_x^i}{r_{31}^i x_W + r_{32}^i y_W + r_{33}^i z_W + t_x^i} \approx \tilde{v} + k_1 \tilde{v} \left(\tilde{u}^2 + \tilde{v}^2\right) \tag{6}$$

where the rotation matrix $\boldsymbol{R}^i$ is specified by independent yaw-pitch-roll angles ϕ_x, ϕ_y and ϕ_z. We will call $\boldsymbol{e}^i = [\phi_x^i, \phi_y^i, \phi_z^i, t_x^i, t_y^i, t_z^i]^T$ be the *extrinsic camera parameters* for viewpoint i.

Having the intrinsic camera parameters, $\boldsymbol{s}$ and the view-dependent extrinsic camera parameters $\boldsymbol{e}^i$, we can always determine the mapping of 3D points in the world coordinate frame into the camera coordinate frame. In order to compute the camera image coordinates (u, v) from the corresponding 3D point (x_W, y_W, z_W) in the world frame,

1. compute the normalized camera image coordinates $(\tilde{u}, \tilde{v})$ by solving the implicit forms of Eqs. (5) and (6) using an appropriate iterative gradient method, say Newton method, with initial values $(\tilde{u}, \tilde{v}) = (u', v')$.

2. compute the actual image coordinates (u, v) from $(\tilde{u}, \tilde{v})$ using Eq. (4).

3 Solution to Hand-Eye Calibration

Most of the previous work on hand-eye calibration did not take into account the lens distortion for hand-eye calibration, since their main interest was visual

servoing for robotic manipulation, which in most cases only utilized central regions of camera images where the pinhole camera might be sufficient to model the 3D-2D point projection.

When we wish to use a camera with small focal length and wider views for 3D object modeling, lens distortion becomes significant for precise 3D measurements, or for acquiring object modeling from multiple view observations. We propose here an efficient and accurate approach which can deal with lens distortion for the camera model as well as precise determination of robotic hand and camera transformations. In our hand-eye calibration strategy, we take the following three steps as shown in Fig. 5: In Step 1, we locate a calibration pattern board at different heights (z_W values) in the world coordinate frame, and then take snapshots of the calibration pattern board from multiple viewpoints by moving the robot end-effector as shown in Fig. 6. We then estimate the intrinsic and extrinsic camera parameters for each viewpoint independently, assuming that intrinsic camera parameters are not necessarily equal. In Step 2, we re-estimate intrinsic and extrinsic camera parameters by imposing the constraint that all intrinsic camera parameters be equal. In Step 3, we compute the homogeneous transformation between the robot tool and the camera coordinate frames as well as that between the robot base and the world coordinate frames.

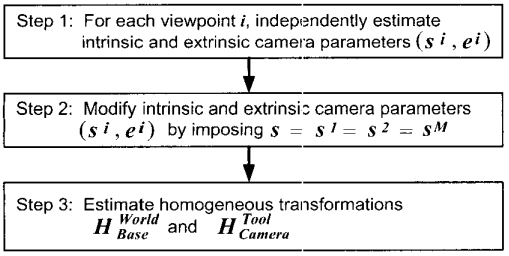

Figure 5: *Calibration steps.*

Step 1: Initial estimation of camera parameters

In Step 1, we generate multiple viewpoints (M viewpoints in total) by moving the camera by the robot and then take a snapshot of the planar calibration pattern board which is located in different heights (z_W) specified in the world coordinate frame. As shown in Fig. 6, the calibration patterns consist of small black circles (N circles in total) and the 3D coordinates of the centroids of the calibration patterns are measured in the world frame. By analyzing the snapshots of the calibration patterns by a computer, we can estimate the 2D image coordinates of the centroids in the camera image frame. For each viewpoint i and each circle j, let (x_j^i, y_j^i, z_j^i) and (u_j^i, v_j^i) be the 3D coordinates of the calibration pattern centroids in the world coordinate frame and the corresponding 2D image coordinates in the camera image frame measured and accumulated from various heights. Then from Eqs. (5) and (6), we

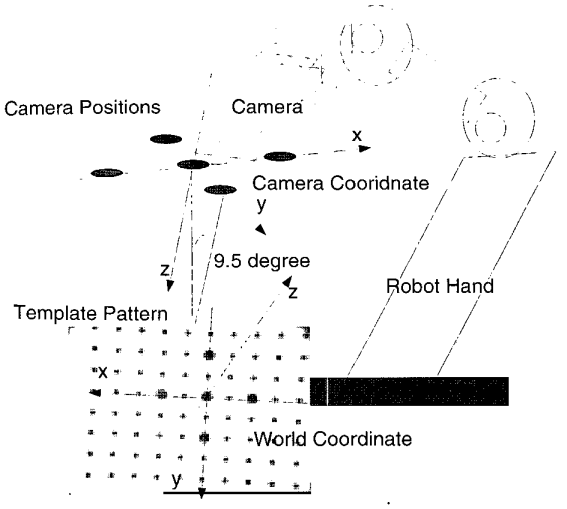

Figure 6: *The imaging viewpoints are illustrated. A single camera is mounted on the robot hand. The five camera positions are located such that the angular distance, as subtended at the object, is between successive positions at 9.5 degrees.*

obtain

$$\frac{r_{11}^i x_W + r_{12}^i y_W + r_{13}^i z_W + t_x^i}{r_{31}^i x_W + r_{32}^i y_W + r_{33}^i z_W + t_x^i} = \tilde{u}_j^i + k_1 \tilde{u}_j^i \left((\tilde{u}_j^i)^2 + (\tilde{v}_j^i)^2\right)$$

$$\frac{r_{21}^i x_W + r_{22}^i y_W + r_{23}^i z_W + t_x^i}{r_{31}^i x_W + r_{32}^i y_W + r_{33}^i z_W + t_x^i} = \tilde{v}_j^i + k_1 \tilde{v}_j^i \left((\tilde{u}_j^i)^2 + (\tilde{v}_j^i)^2\right)$$

$$\tilde{u}_j^i = \frac{u_j^i - u_0}{\alpha_u} \quad \tilde{v}_j^i = \frac{v_j^i - v_0}{\alpha_v}. \tag{7}$$

In Step 1, we first estimate intrinsic camera parameters $\boldsymbol{s}^i = [\alpha_u^i, \alpha_v^i, u_0^i, v_0^i, k_1^i]^T$ and extrinsic camera parameters $\boldsymbol{e}^i = [\phi_x^i, \phi_y^i, \phi_z^i, t_x^i, t_y^i, t_z^i]^T$ for different viewpoint i independently. To do this, we apply Weng's algorithm based on a non-linear iterative algorithm[11]. Due to the space limit of this paper, we will not explain the details of this algorithm. But the main part of the algorithm is that this problem is converted into an optimization problem which minimizes the objective function of image registration error between 2D measurement coordinates (u_j^i, v_j^i) and the 2D projected coordinates $(\bar{u}_j^i, \bar{v}_j^i)$ based on the parametric representation of intrinsic and extrinsic camera models $\boldsymbol{s}^i$ and $\boldsymbol{e}^i$ computed from Eqs. (7).

The objective function is actually defined by

$$f = \sum_j \{(u_j^i - \bar{u}_j^i)^2 + (v_j^i - \bar{v}_j^i)^2\} \tag{8}$$

which should be minimized with respect to $\boldsymbol{s}^i$ and $\boldsymbol{e}^i$.

Step 2: Integration of camera parameter estimates from multiple views

In Step 1, intrinsic camera parameters $\boldsymbol{s}^i$ (i = 1, 2, ..., M) are not necessarily equal, since the algorithm independently estimates these parameters for viewpoint i. In Step 2, we integrate all estimates to obtain an optimal estimate of intrinsic and extrinsic camera parameters $\boldsymbol{s}$ and $\boldsymbol{e}^i$ by a non-linear iteration algorithm. As for the initial estimates for the iteration, we use the estimates $\boldsymbol{s}^0$ and $\boldsymbol{e}^i$ obtained in Step 1. More specifically, we attempt to minimize f in Eq. (8) under the additional constraints:

$$\boldsymbol{s} = \boldsymbol{s}^1 = \boldsymbol{s}^2 = ... = \boldsymbol{s}^M \tag{9}$$

We also apply an iterative technique of non-linear optimization to attain a robust and accurate estimate. In this case, we have a good initial estimate for $\boldsymbol{s}$ and $\boldsymbol{e}^i$ ($i = 1, ..., M$), and then we apply an extended Kalman filter-based updating scheme [5] so that sequential updating of parameters can be attained along with outlier elimination. The sequential updating is useful when some outliers due to the noise exist. In our case, we have a large number of measurements of image calibration points (u_j, v_j) ($j = 1, 2, ..., N$) where N is typically around 1000. The extended Kalman filter based updating can be implemented for a small number of constraints in each iteration of updating; more specifically only two degrees of freedom for the constraints are necessary. Note that parameters to be estimated are $(\boldsymbol{s}, \boldsymbol{e}^1, \boldsymbol{e}^2, ..., \boldsymbol{e}^M)$ and the dimension of the parameter space is $5 + 6M$. The constraint equation acquired from Eq. (9) is simply two-dimensional associated with the image measurements (u_j, v_j) for each sequential updating. This reduction of dimensionality greatly helps the reduction of computational complexity.

We also like to mention here that our approach to optimizing intrinsic parameters is important. Most previous approaches did not even consider the optimal estimates of the intrinsic and extrinsic camera parameters for multiple viewpoint cases. In other words, the previous methods simply chose one of the estimates obtained from multiple solutions. This caused a large amount of estimation error in hand-eye calibration especially when the lens distortion was not taken into account.

Step 3: Hand-eye calibration

Once the experimental process has provided the camera parameters $\boldsymbol{s}$ and $\boldsymbol{e}^i$ ($i = 1, 2, ..., M$), the relationship between several coordinates in Fig. 4 will be obtained as each component transformation matrix such as $\boldsymbol{H}^{Tool}_{Camera}$, $\boldsymbol{H}^{Base}_{Tool}$, $\boldsymbol{H}^{World}_{Base}$. Note that $\boldsymbol{H}^{World}_{Camera_i}$ is derived from the extrinsic camera parameters $\boldsymbol{e}^i$. The transitional relationship between those transformation matrices is formulated as the following equation:

$$\boldsymbol{H}^{World}_{Camera_i} = \boldsymbol{H}^{Tool_i}_{Camera_i} \boldsymbol{H}^{Base}_{Tool_i} \boldsymbol{H}^{World}_{Base} \tag{10}$$

Since one camera is fixed in the tool position in our camera setting, the images of different viewpoints are produced by tool position control. Given the positioning control Puma commands $XYZOAT$ in Eq. (18), we can specify the transformation matrix $\boldsymbol{H}^{Base}_{Tool_i}$ [4]. The two unknown transformation matrices in Eq. (10) are $\boldsymbol{H}^{World}_{Base}$ and $\boldsymbol{H}^{Tool_i}_{Camera_i}$. Note that since the transformation from Tool coordinates to Camera coordinates is viewpoint independent, we can denote an i invariant matrix by $\boldsymbol{H}^{Tool}_{Camera}$. Using the results of multiple camera calibration results, we solve $\boldsymbol{H}^{World}_{Base}$ first, use this matrix, and then compute the last unknown matrix $\boldsymbol{H}^{Tool}_{Camera}$. Since we approximately know that the world z-axis and robot base z-axis are almost parallel, $\boldsymbol{H}^{World}_{Base}$ is almost equal to an identity matrix in a rotational part, some values in a translational part. For simplicity, we show the two pair viewpoints case among five viewpoints, denoted i and j for the transitional relationship in Eqs. (11):

$$\boldsymbol{H}^{World}_{Camera_{i/j}} = \boldsymbol{H}^{Tool}_{Camera} \boldsymbol{H}^{Base}_{Tool_{i/j}} \boldsymbol{H}^{World}_{Base} \tag{11}$$

We eliminate $\boldsymbol{H}^{Tool}_{Camera}$ from i and j Eqs. (11), and obtain:

$$\boldsymbol{A}_{ij}\, \boldsymbol{X} - \boldsymbol{X}\, \boldsymbol{B}_{ij} = \boldsymbol{O} \tag{12}$$

where those notations are provided as:

$$\boldsymbol{X} = \boldsymbol{H}^{World}_{Base} \tag{13}$$

$$\boldsymbol{A}_{ij} = (\boldsymbol{H}^{Base}_{Tool_i})^{-1} \boldsymbol{H}^{Base}_{Tool_j} \tag{14}$$

$$\boldsymbol{B}_{ij} = (\boldsymbol{H}^{World}_{Camera_i})^{-1} \boldsymbol{H}^{World}_{Camera_j} \tag{15}$$

Although an analytical derivation of solutions to $\boldsymbol{X}$ in Eq. (12) is generally tough [8, 2], we can derive a simple iterative approach for Eq. (12) in the following form: Let ψ_x, ψ_y, and ψ_z be the yaw-pitch-roll angles associated with the homogeneous transformation $\boldsymbol{H}^{World}_{Base}$, and (p_x, p_y, p_z) be the translational vector associated with the transformation. Then

$$\boldsymbol{H}^{World}_{Base} = \begin{bmatrix} c_z c_y & c_z s_y s_x - s_z c_x & c_z s_y c_x + s_z s_x & p_x \\ s_z c_y & s_z s_y s_x + c_z c_x & s_z s_y c_z - c_z s_x & p_y \\ -s_y & c_y s_x & c_y c_x & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}$$

We apply the Broyden-Fletcher-Goldfarb-Shanno optimization algorithm [1] to obtain the solution of $\boldsymbol{q} = (\psi_x, \psi_y, \psi_z, p_x, p_y, p_z)^T$ which minimizes the objective function:

$$f(\boldsymbol{q}) = \sum_{i,j(i \neq j)} ||\boldsymbol{A}_{ij}\, \boldsymbol{X} - \boldsymbol{X}\, \boldsymbol{B}_{ij}||^2. \tag{17}$$

For this non-linear iteration method, we can appropriately select an initial estimate of $\boldsymbol{q}$, since we approximately know the position of the robot base with respect to the world coordinate frame. But of course,

our initial estimate may not be sufficiently close for us to apply a simple gradient-descent method like Newton's. Therefore we utilize Broyden-Fletcher-Goldfarb-Shanno optimization method whose convergence is known to be stable.

Note that estimation of $\boldsymbol{H}_{Camera}^{Tool}$ is derived as the same form Eq. (12) in the case of $\boldsymbol{H}_{Base}^{World}$. When the robot tool is moving toward a specific position, we have the robot positioning control parameter $XYZOAT$ and its corresponding homogeneous transformation $\boldsymbol{H}_{Tool_i}^{Base}$. The three transformation matrices $\boldsymbol{H}_{Camera}^{Tool}$, $\boldsymbol{H}_{Tool_i}^{Base}$, $\boldsymbol{H}_{Base}^{World}$ will produce the desired transformation matrix $\boldsymbol{H}_{Camera_i}^{World}$ by Eq. (10). Therefore, we can now generate both intrinsic and extrinsic camera parameters for any tool positions if we have the end-effector positioning parameters $XYZOAT$. The output of our solution is the optimized intrinsic camera parameters $\boldsymbol{s}$ and the computed extrinsic camera parameters $\boldsymbol{e}$ at any arbitrary tool position. This has eliminated the need for re-calibration as the robot end-effector is moved to different positions.

4 Viewpoint Generation by Robotic Motion

Our application aim of calibration is to generate viewpoints and to obtain all parameters for transformations associated with viewpoints. In order to establish this task, we need a kinematics to locate the camera at arbitrary positions and orientations in the world coordinate frame by moving the end-effector of the robot. So the problem is specified as follows:

- Given a homogeneous transformation from the world coordinate frame to the camera coordinate frame, generate the robot motion commands.

In our current system setup, we use Puma robot hand whose end-effector tool position is controlled by the six parameters $XYZOAT$ and is expressed in terms of the homogeneous transformation matrix:

$$\boldsymbol{H}_{Base}^{Tool} = \begin{bmatrix} c_O s_T - s_O s_A c_T & c_O c_T + s_O s_A s_T & s_O c_A & X \\ s_O s_T + c_O s_A c_T & s_O c_T - c_O s_A s_T & -c_O c_A & Y \\ -c_A c_T & c_A s_T & -s_A & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

Given $\boldsymbol{H}_{Base}^{Tool}$, $XYZOAT$ parameters are computed by each element of Eq. 18 [4]. So our viewpoint generation problem is how to compute $\boldsymbol{H}_{Base}^{Tool}$, using $\boldsymbol{H}_{World}^{Camera}$ based on extrinsic camera parameters $\boldsymbol{e}$. This can be done by computing

$$\boldsymbol{H}_{Base}^{Tool} = \boldsymbol{H}_{Base}^{World} \boldsymbol{H}_{World}^{Camera} \boldsymbol{H}_{Camera}^{Tool} \quad (19)$$

Obviously, $\boldsymbol{H}_{Base}^{World}$ and $\boldsymbol{H}_{Camera}^{Tool}$ are determined by the camera calibration discussed in the previous section.

5 Applications

5.1 Application to Object Modeling from Multiple Views

The estimated intrinsic camera parameters $\boldsymbol{s}$ and extrinsic camera parameters $\boldsymbol{e}$ at the generated viewpoints are used to compute feature coordinates for an object modeling. In the model acquisition phase of our project [7], the feature correspondence in the multiple views is handled by a human operator with reference to the automatically extracted feature candidates and epipolar line. The number of viewpoints are five, with one taken straight from above, and two each from viewpoints that are offset along the X and Y world coordinates. After the feature correspondences in multiple views are established, the feature distribution in the 3D world coordinate frame is computed using the camera parameters $\boldsymbol{s}$ and $\boldsymbol{e}$ by least-mean-square minimization as described in [6]. An example of an polyhedral object model is illustrated in Fig. 3. The computed 3D feature points in this model are displayed in the wireframe representation. In each surface, feasible Grasping coordinates are also introduced by human operator's decision whether the robot should grasp that surface or not. For example, the object model in Fig. 3 has two Grasping coordinates, which are used for robot manipulation of grasping.

5.2 Application to Object Localization and Grasping by Robot Motion

Our final goal of the research is to develop a vision-based bin-picking system for automation. Once 3D models are generated through the human-assisted model acquisition system, the wireframe-based CAD model will be used for the robot to recognize and grasp the object which will be randomly located in a bin in the robotic workspace. In our strategy, the robot initially takes two views of the object whose viewpoints are determined by the same method described in Section 4. The robot then attempts to recognize the object based on the correspondence between the model features in the 3D CAD model and image features extracted from camera images taken at such viewpoints. In our current setup, we utilize point features and line features for model and image features. We apply the model-based stereo vision algorithm developed by Kosaka and Kak [6] to extract image features and to recognize and localize objects in the world coordinate frame.

If the system generates multiple hypotheses for object poses from original two views, the system attempts to verify or reject these hypotheses by moving the camera to an additional viewpoint. This camera motion is automatically generated by the robot motion so as to disambiguate the initial set of hypothesis. A preferable verification viewpoint is selected such that the camera is pointing out to the normal direction of the localized top surface because, in our application, the robot hand picks up the object along the axis of that perpendicular direction. This greatly improves the pose accuracy with minimum computations by disambiguating the correspondence problem with a grasping application.

After the object pose is fully verified by the system,

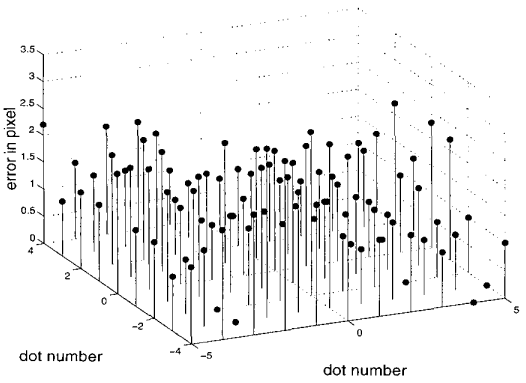

Figure 7: *Error of calibration parameters with pixel deviation between the projected value and the actual image point. The average image registration error is* 1.38 ± 0.79 *pixel/dot.*

we obtain the homogeneous transformation $\boldsymbol{H}_{World}^{Object}$ from the object coordinate frame to the world coordinate frame. Based on the 3D pose of the object, the system now generates the grasping strategy by computing the homogeneous transformation $\boldsymbol{H}_{Object}^{Tool}$ from the tool coordinates to the object coordinates. Finally, the robot positioning command will be generated by computing $\boldsymbol{H}_{Base}^{Tool}$ as follows:

$$\boldsymbol{H}_{Base}^{Tool} = \boldsymbol{H}_{Base}^{World} \boldsymbol{H}_{World}^{Object} \boldsymbol{H}_{Object}^{Tool} \quad (20)$$

6 Experimental Results

6.1 Experiments of Calibration Accuracy

We implemented the off-line proposed calibration process in a SUN Workstation ULTRA 10. It takes several minutes of computation to optimally estimate the camera parameters and homogeneous transformation matrices such as $\boldsymbol{s}$, $\boldsymbol{H}_{Camera}^{Tool}$, $\boldsymbol{H}_{Base}^{World}$. In this experiment, we used a Sony $DC-47$ monocular $1/3$ *inch* CCD camera with Pulnix lens of focal-length 16 *mm*. And then, as an on-line process, we implemented to construct $\boldsymbol{s}$ and $\boldsymbol{e}^i$ at a viewpoint i in a PC (Pentium $350MHz$) with a PUMA 761 controller. The whole computation time is less than $0.2sec$ to generate a calibration matrix as the tool moves to one position, which is less than the time of servoing the Puma hand. Fig. 1 shows the result of the error evaluation of the estimated calibration parameters with lens distortion, by superimposing the white cross-bars projected with the estimated parameters. Fig. 7 shows the quantitative deviation distribution between the white-bars and the actual circular calibration patterns in the image. We evaluated the results of estimating transformation matrices by comparing other solutions in Table 1, which included the method without lens distortion and a simple linear interpolation. Obviously, the modeling with lens distortion provides the outstanding accuracy of registration.

Table 1: *Error of image registration.*

Method	Average Error (pixel)
our method with distortion	1.38 ± 0.79
our method without distortion	1.71 ± 0.91
interpolation method	4.21 ± 2.1

6.2 Application Experiments of Object Localization and Grasping

In order to verify the applications of our calibration accuracy, we made experiments of object localization and grasping tasks described in Subsection 5.2. The system generates Puma gripper positioning parameters $XYZOAT$ to generate the desired viewpoints for capturing the images of objects shown in Fig. 8. The system first extracts image features of vertices to recognize and localize objects in the scene. Figs. 8 (a) and (b) represent Canny's edge images for two viewpoints (left and right views) and corner points of image features are extracted from the Canny's edge images. The next task is to establish a correspondence between the model and image features. As we have discussed in Subsection 5.1, the model features are registered in a CAD model based on the human-assisted model acquisition system. We then applied the model-based stereo vision algorithm [6] which generates multiple pose hypotheses from candidate correspondences between model features and image features.

In order to verify and/or reject the pose hypotheses, a new viewpoint for camera observation is computed so that the desired images can be obtained at that viewpoint. If the hypothesis is not fully verified, the system will try to capture an additional image from the next viewpoint, and the object pose is recomputed on the basis of the model-image feature correspondence candidates newly obtained.

Fig. 8 (c) shows the image taken at the new viewpoint for verification. Fig. 8 (d) depicts the wireframe object model which is selected by the object recognition. Fig. 8(e) demonstrates the result of the pose estimation. In this figure, the wireframe of the object model based on the estimated pose is superimposed onto the original image frame taken at the verified viewpoint. Therefore, this figure demonstrates the accuracy of our recognition and pose estimation. Finally Fig. 8(f) shows that the robot hand successfully picks up the object based on the verified 3D pose.

7 Conclusions

This paper presented a new method called "SmartView" of hand-eye robotic calibration for vision-based object modeling and grasping. Our method provided a 1.0-pixel level of image registration accuracy when a standard Puma robot generates arbitrary viewpoints. In order to attain this accuracy, we formalized hand-eye calibration with the lens distortion camera model in addition to a simple pin-hole model. Also, our formalism also included the optimal integration of camera parameter estimates computed from multiple views. Through the modeling and manipulation experiments using a Puma robot,

we demonstrated the accuracy and performance of our hand-eye calibration method, and showed that our vision system had sufficient power for active viewpoint generation for accurate 3D model acquisition and robust 3D object recognition. Currently, we are working on a quantitative evaluation of the robustness of our method.

Acknowledgment

The support of A.M.T.D., Ford Motor Co.is gratefully acknowledged.

References

[1] E. K. P. Chong and S. H. Zak, *An Introduction to Optimization*, pp. 147-165, John Wiley & Sons, Inc., New York, 1996.

[2] F. Dornaika and R. Horaud, "Simultaneous Robot-world and Hand-eye Calibration," IEEE Transactions of Robotics and Automation," Vol. 14, No. 4, pp.617-621, Aug. 1998.

[3] O. D. Faugeras and M. Herbert, Representation, Recognition, and Locating of 3-D Objects, *International Journal of Robotics and Research*, Vol. 5, No. 3, pp. 27-52, 1986.

[4] K. S. Fu, R. C. Gonzales, and C. S. G. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*, pp. 12-81, McGraw-Hill, New York, 1987.

[5] A. Kosaka and A. C. Kak, Fast Vision-guided Mobile Robot Navigation using Model-based Reasoning and Prediction of Uncertainties, *Computer Vision, Graphics, and Image Processing - Image Understanding*, Vol. 56, No. 3, pp. 271-329, 1991.

[6] A. Kosaka and A. C. Kak, Stereo Vision For Industrial Applications, *Handbook of Industrial Robotics, edit.* S. Y. Nof, pp. 269-294, John Wiley & Sons, Inc., New York, 1999.

[7] Y. Motai, P. Wilson, and A. C. Kak, Learning by Showing for Robot Vision of the Future, *Robot Vision Laboratory Technical Report*, School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana, Nov., 1999.

[8] Y. C. Shiu and S. Ahmad, Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form $AX = XB$, pp. 16-29, *IEEE Transactions on Robotics and Automation RA-5*(1), Feb., 1989.

[9] K. Tarabanis, P. K. Allen and R. Y. Tsai, Survey of Sensor Planning in Computer Vision, pp. *IEEE Transactions on Robotics and Automation RA-11*(1), Feb., 1995.

[10] R. Y. Tsai, A versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses, pp. 323-344, *IEEE Transactions on Robotics and Automation RA-3*(4), 1988.

[11] J. Weng, P. Cohen, and M. Herniou, Camera Calibration with Distortion Models and Accuracy Evaluation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 965-980, 1992.

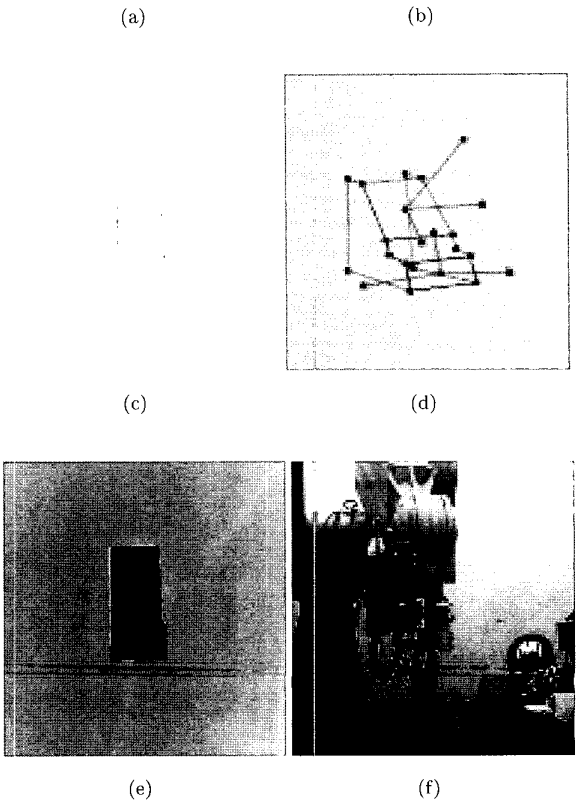

Figure 8: *Application Experiments: (a) (b) Canny's edge maps representing image features used to generate pose hypotheses for (a) left and (b) right initial viewpoints, (c) edge map from another viewpoint to verify the hypotheses, (d) wireframe model of the object generated by the human-assisted model acquisition system, (e) superposition of the wireframe object model onto the original image using the estimated pose, (f) robot manipulation.*