Proceedings of the 1992 IEEE
International Conference on Robotics and Automation
Nice, France - May 1992

# 3D Relative Position And Orientation Estimation Using Kalman Filter For Robot Control

Jiang Wang William J. Wilson

Dept. of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario N2L 3G1

## Abstract

A vision based position sensing system which provides three-dimensional (3D) relative position and orientation (pose) of an arbitrary moving object with respect to a camera for a real-time tracking control is studied in this paper. Kalman filtering is applied to vision measurements for the implicit solution of the photogrametric equations and to provide significant temporal filtering of the resulting motion parameters resulting in optimal pose estimation. Both computer simulation and real-time experimental results are presented to verify the effectivenss of the Kalman filter approach with large vision measurement noise.

## 1 Introduction

One of the central problems in vision guided robot tracking control is the determination of the relative position and orientation (pose) of a randomly moving object with respect to the robot end-effector mounted camera in three-dimensional (3D) space. Many approaches have been proposed to estimate the 3D relative position and orientation using a single image view. The common approachs used are based on one of two methods. One method is to use task specific image preprocessing to extract the image feature location measurements which are combined with known object CAD descriptions to estimate pose parameters for effective dynamic tracking control. Another method is to use general scene analysis to identify objects and to determine the pose parameters. The second method requires considerable preprocessing so it is difficult to perform these computations at a rate suitable for real-time dynamic control. Photogrammetric techniques[10] and 2D to 3D line or point correspondence techniques[3] [2] [7] are generally used in the first method, however they need accurate image measurements.

In the manufacturing environment, low-cost camera systems are commonly used for many tasks. This kind of vision measurement includes significant noise due to the camera's characteristics, image signal spatial quantization and amplitude discretization, lens distortion, sensor pixel level errors, etc. Even for high-cost camera systems, these error sources cannot be totally avoided. Noisy image can result in poor individual pose estimates. To reduce the effect of image measurement noise on the pose estimates, a time based filter can be applied to filter out some of noise when a sequence of images are used for tracking control. Several authors have applied filter theory to estimate pose parameters from a sequence of noisy images. Chang et al[4] presented an approach to estimate 3D motion parameters for target tracking using a Kalman filter. But their method was based on the assumption that the target is distant and it can be considered as a single feature point. Therefore, the motion parameters do not include the orientation parameters which are difficult to estimate. Broida and Chellappa[1] have presented a dynamic model to estimate 3D motion parameters using a Kalman filter. However, they confined their discussion to planar object motion, and the 3D problem was far from solved. Dickmanns et al[5] have applied Kalman filter theory to do state estimation for the motion of object using image sequence processing for guidance of autonomous vehicles. Wilson[9] also presented an approach to estimate 3D motion parameters for tracking control using a Kalman filter. This approach has been demonstrated to work well for a planar object motion in 2D tracking control[6][8]. This paper presents the extension of this approach for estimating 3D motion parameters for 3D tracking control. Both computer simulation and real-



0-8186-2720-4/92 $3.00 ©1992 IEEE

time experiments are used to validate the performance.

In this approach, an end-point mounted camera and image preprocessor system provide image plane measurements of selected known task object features. The Kalman filter uses these measurements for the implicit solution of the photogrammetric equations and for filtering of the resulting motion parameters to give optimal pose estimation for real-time tracking control. The experimental results show that this approach not only reduces the effect of large measurement noise, but also has several other advantages. The Kalman filter predicts the pose parameters at the next time step which can be used to predict the image plane feature point locations. This allows the vision sensor system to process only small window areas in the image plane to obtain the required feature point measurements. This "directed" image processing approach results in a significant reduction in image processing time. The Kalman pose predictions also help in overcoming the unique ness problem of the pose solution, which usually exists in the single image view.

This paper consists of five sections. Section 2 presents the Kalman filter algorithm to estimate 3D pose parameters and motion parameters. Section 3 discusses the experimental system implementation. Section 4 presents the computer simulation analysis and the real-time experimental results. Conclusions are given in Section 5.

## 2 3D POSE Estimation Using the Extended Kalman Filter

This section first presents the mathematical relationship between the pose parameters (the states) and the image plane feature measurements, and then presents the Kalman filter algorithm for optimal pose estimation.

### 2.1 Feature Point Relationships

The geometry of the problem is illustrated in Figure 1. The nomenclature used is as follows:

| | |
|---|---|
| $O^o - X^oY^oZ^o$ | Object reference frame, |
| $O^c - X^cY^cZ^c$ | Camera reference frame with $Z^c-$ axis coinciding with the optical axis, |
| $O^i - X^iY^i$ | Image reference frame, |
| $F$ | Camera focal length, |

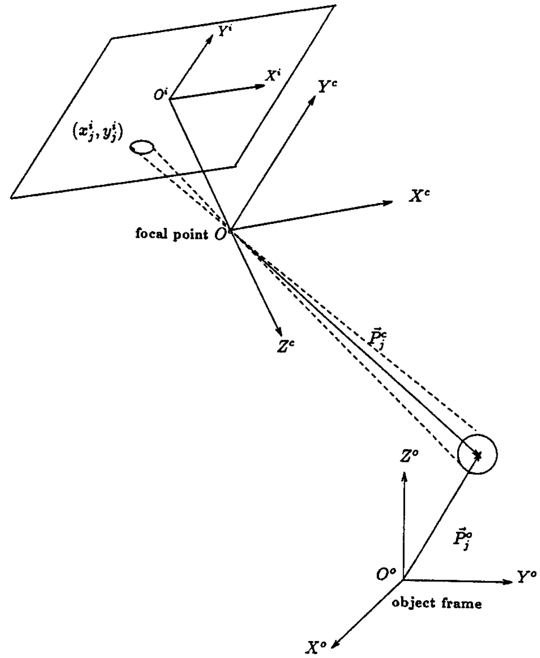


Figure 1. Object Feature Image Projection

| | |
|---|---|
| $\vec{P}_j^o = (X_j^o, Y_j^o, Z_j^o)^T$ | Coordinate vector of the $j$th object feature point in the object frame, |
| $\vec{P}_j^c = (X_j^c, Y_j^c, Z_j^c)^T$ | Coordinate vector of the $j$th object feature point in the camera frame, |
| $x_j^i, y_j^i$ | Corresponding $j$th projection feature point coordinates in the image frame, |
| $P_X, P_Y$ | Inter-pixel spacing along the $x^i$ and $y^i$ axes $(m/pixel)$, |
| $\vec{T} = [X, Y, Z]^T$ | Relative position vector, |
| $\phi, \alpha, \psi$ | Relative *Roll*, *Pitch* and *Yaw* orientation parameters. |

The transformation of the coordinates of $j$th feature point in the object frame to its coordinates in the camera frame can be written as

$$\vec{P}_j^c = \vec{T} + R(\phi, \alpha, \psi)\vec{P}_j^o \tag{1}$$

where $\vec{P}_j^o$ is assumed to be known from the CAD model and $R(\phi, \alpha, \psi)$ is the rotation matrix

$$R(\phi,\alpha,\psi) =$$

$$\begin{bmatrix} C_\phi C_\alpha & C_\phi S_\alpha S_\psi - S_\phi C_\psi & C_\phi S_\alpha C_\psi + S_\phi S_\psi \\ S_\phi C_\alpha & S_\phi S_\alpha S_\psi + C_\phi C_\psi & S_\phi S_\alpha C_\psi - C_\phi S_\psi \\ -S_\alpha & C_\alpha S_\psi & C_\alpha C_\psi \end{bmatrix}$$

where $\quad C_\alpha = \cos\alpha \qquad S_\alpha = \sin\alpha$

For the point projection camera model shown in Figure 1, the corresponding projection feature point locations in the image frame are obtained by

$$\begin{aligned} x_j^i &= -\frac{FX_j^c}{P_X Z_j^c} \\ y_j^i &= -\frac{FY_j^c}{P_Y Z_j^c} \end{aligned} \tag{2}$$

Combining Eq. (1) and Eq. (2), results in the equations defining the image feature locations as nonlinear functions of the pose parameters[9]. However, directly solving these nonlinear equations is complicated and needs accurate image measurements. Since the camera focal length is often small with respect to the $Z$ parameter, relatively large changes in some pose parameters may cause small changes in the image feature point locations, especially for coplanar and closely spaced object features. Therefore, some pose parameters have high sensitivities to the small image measurement changes. Measurement errors can result in very poor individual pose estimates when the nonlinear equations are solved directly. The next section presents a Kalman filter approach, which indirectly solves the nonlinear equations and filters out measurement noise to provide a time series of optimal pose estimates based on the sequence of image measurements used in the tracking problem.

## 2.2 Kalman Filter Algorithm

The Kalman filter requires two system models; a dynamic model and an output model. This section first reviews the two models presented in [9] and then presents the output equations for the 3D problem, followed by the Kalman filter algorithm.

A dynamic model is defined to describe the relative object motion with respect to the camera in terms of the system states related to the pose parameters. Since arbitrary relative motion cannot be described by a structured dynamic model, an approximate dynamic model is considered based on the assumption that the relative motion in a 3D space is slow and smooth enough, such that the first derivatives of the pose parameters $(\dot{X}, \dot{Y}, \dot{Z}, \dot{\phi}, \dot{\alpha}, \dot{\psi})$ can be considered constant over a sample period. The effects of their higher order derivatives may then be considered as disturbance noise added to the dynamic model.

Let $W = [X, \dot{X}, Y, \dot{Y}, Z, \dot{Z}, \phi, \dot{\phi}, \alpha, \dot{\alpha}, \psi, \dot{\psi}]^T$ be the system state vector; $\gamma$ be the disturbance noise vector of the dynamic model, which is assumed to be described by a zero mean and Gaussian noise with covariance $Q$; $T$ be the sample period; and $k$ be the sample time step. The discrete time system dynamic model is then expressed by

$$W_k = AW_{k-1} + \gamma_k \tag{3}$$

where

$$A = \begin{bmatrix} 1 & T & & & & & \\ 0 & 1 & & & & & \\ & & 1 & T & & & \\ & & 0 & 1 & & & \\ & & & & \ddots & & \\ & & & & & 1 & T \\ & & & & & 0 & 1 \end{bmatrix}_{12\times 12}$$

The output model is defined by the relationship between the vision measurements $Z_k$ of the image feature point locations and the system states $W_k$. As described in the section 2.1, this relationship includes 3D coordinate transformation (Eq.1) and projection transformation (Eq. 2). For the dynamic model (Eq. 3), at least six independent measurement equations, which can be formed using three non-colinear object features, are required such that the system states are totally observable. Let $(X_j^o, Y_j^o, Z_j^o)$, $j = 1,2,3$ be coordinates of three known non-colinear object feature points in the object frame, and $(x_j^i, y_j^i)_k$ be $k$th time measurements of the corresponding feature image locations. The Kalman output model is then obtained from Eq. 1 and 2 as

$$Z_k = G(W_k) + \nu_k \tag{4}$$

where

$$\begin{aligned} Z_k &= [x_1^i,\ y_1^i,\ x_2^i,\ y_2^i,\ x_3^i,\ y_3^i]_k^T \\ G(W_k) &= -F\left[\frac{X_1^c}{P_X Z_1^c}, \frac{Y_1^c}{P_Y Z_1^c}, \frac{X_2^c}{P_X Z_2^c}, \frac{Y_2^c}{P_Y Z_2^c}, \frac{X_3^c}{P_X Z_3^c}, \frac{Y_3^c}{P_Y Z_3^c}\right]_k^T \\ X_j^c &= X + C_\phi C_\alpha X_j^o + (C_\phi S_\alpha S_\psi - S_\phi C_\psi)Y_j^o \\ &\qquad + (C_\phi S_\alpha C_\psi + S_\phi S_\psi)Z_j^o \\ Y_j^c &= Y + S_\phi C_\alpha X_j^o + (S_\phi S_\alpha S_\psi + C_\phi C_\psi)Y_j^o \\ &\qquad + (S_\phi S_\alpha C_\psi - C_\phi S_\psi)Z_j^o \\ Z_j^c &= Z - S_\alpha X_j^o + C_\alpha S_\psi Y_j^o + C_\alpha C_\psi Z_j^o \end{aligned}$$

$\nu_k$ is measurement noise which is assumed to be described by a zero mean Gaussian noise with covariance $R$.

Since a 2D image provides weaker depth information, three feature measurements may not provide enough information needed for good estimation of $Z$, $Pitch$ and $Yaw$ parameters. For the output model (Eq. 4), additional features can be included by extending the output equation with no change in the form of the model. Also, for other vision sensor models, the form of output model is the same with only a change in the output function $G(W_k)$.

The Kalman filter algorithm normally requires a linear dynamic model and a linear output model. Since the output model is nonlinear, the extended Kalman filter must be used. The extended Kalman filter acts as a position sensor which receives image measurements and provides feedback pose parameters at every sample time step for tracking control.

The recursive equations defining the extended Kalman filter algorithm are given by the following:

Prediction Equation:

$$\hat{W}_{k,k-1} = A\hat{W}_{k-1,k-1} \tag{5}$$

$$P_{k,k-1} = AP_{k-1,k-1}A^T + Q_{k-1} \tag{6}$$

Linearization:

$$H_k = \frac{\partial G(W)}{\partial W}\Big|_{W=\hat{W}_{k,k-1}} \tag{7}$$

Kalman Gain:

$$K = P_{k,k-1}H_k^T(R_k + H_kP_{k,k-1}H_k^T)^{-1} \tag{8}$$

Estimate Update:

$$\hat{W}_{k,k} = \hat{W}_{k,k-1} + K(Z_k - G(\hat{W}_{k,k-1})) \tag{9}$$

$$P_{k,k} = P_{k,k-1} - KH_kP_{k,k-1} \tag{10}$$

In the solution of the Kalman filter, it should be noted that proper selection of the input dynamic disturbance noise covariance matrix $Q$ and the measurement noise covariance matrix $R$ is very important. For the defined classes of object features, the image measurement noise distribution can be pre-determined through experiments. However, the selection of $Q$ matrix is nontrivial because the relative object motion is arbitrary. Since the dynamic disturbance noise represents the effects of neglected higher order derivative terms of each pose parameter, the weights in $Q$ should be selected larger for faster motion parameters and smaller for slower motion parameters. Section 4 has more discussion on this topic.

# 3 Implementation

The experimental system consists of four subsystems; a CRS Plus Robot System, a Vision Sensor System(camera and image preprocessor), a Transputer Network and a Master Computer(IBM AT).

The master computer is interfaced to the image preprocessor, the robot controller, and the transputer network. It acts mainly as a communication node for the other systems. The transputer network provides a high speed parallel processing environment for real-time Kalman filter and control processes. The image preprocessor is used to provide the current object feature points' image location measurements to the Kalman filter at rate of 61 Hz. The Kalman filter estimates the current pose parameters of the moving object with respect to the camera, and predicts the pose parameters and object feature point locations at the next sample time, allowing the new window locations to be defined. The new window areas are used by the image preprocessor to extract feature measurements. The pose estimates from Kalman filter are compared with the reference pose to generate the pose error, which is used by the 3D dynamic controller to create the robot joint motor control signals for tracking. These control signals are applied to the reference input of the robot joint servo loops to control the robot motion.

Table 1. Environment Parameters

| | | |
|---|---|---|
| focal length | $F$ | 0.01723 *meter* |
| pixel spacing along $x$ | $P_X$ | 0.00006 $m$ / *pixel* |
| pixel spacing along $y$ | $P_Y$ | 0.00006 $m$ / *pixel* |
| sample period | $T$ | 0.0164 second |
| measurement noise variance | | 0.06 $pixel^2$ |

# 4 Discussion of Simulation And Experimental Results

Both computer simulations and real-time experiments were performed to demonstrate the validity and performance of the Kalman Filer approach.

In the computer simulation, image generation, feature extraction, and 3D relative object motion are implemented along with the Kalman filter algorithm. To make comparisons with experimental results easier the environment parameters, including camera characteristics parameters, sample frequency and image noise level, were selected to correspond with those of the experimental system. These parameters are shown in Table 1. The test relative object motion trajectories are shown in Figure 2. The measurement noise covariance

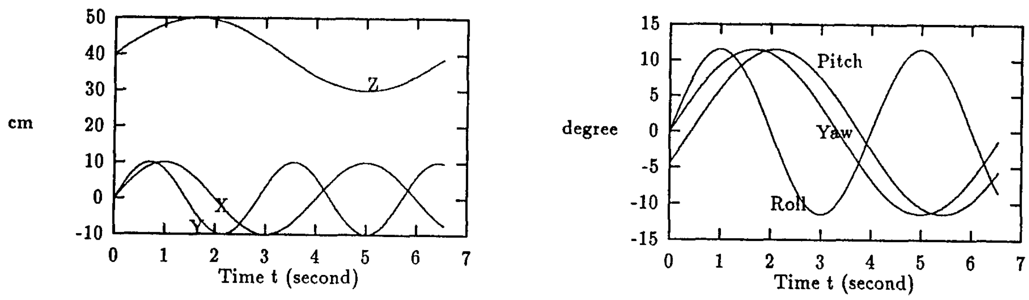


**Figure 2 Object Relative Motion Trajectories In Simulation**

matrix $R$ was selected based on the measurement noise variance.

$$R = diag(0.06,\ 0.06,\ \cdots\cdots\ 0.06)$$

For the dynamic disturbance noise, it is assumed that all disturbance noise is associated with the velocity states $(\dot{X}, \dot{Y}, \dot{Z}, \dot{\phi}, \dot{\alpha}, \dot{\psi})$. Hence, the weights in $Q$ corresponding to the position states $(X, Y, Z, \phi, \alpha, \psi)$ were selected to be zero. The weights corresponding to the velocity states in $Q$ were obtained by calculating the variance of the velocity errors between the samples. This was feasible since the relative parameter trajectories in the simulation are known.

Figure 3 shows the Kalman tracking errors for the true relative motion trajectories shown in Figure 2 when using five non-coplanar object features. It can be seen that the Kalman tracking accuracies of $X$ and $Y$ are within $\pm 0.3$ *mm*, $Z$ within $\pm 0.6$ *mm*, *Roll* within $\pm 0.1$ *degree*, and *Pitch* and *Yaw* within $\pm 0.4$ *degree*. The accuracies of parameters $X$, $Y$ and *Roll* are better than those of parameters $Z$, *Pitch* and *Yaw*. This is because the 2D image is more sensitive to changes in parameters $X$, $Y$ and *Roll* than changes in depth parameters $Z$, *Pitch* and *Yaw*.

Table 2 shows image measurement error variances and Kalman estimate output error variances. The Kalman estimate output errors are the errors between the true image feature locations and the calculated image feature locations from Kalman estimates. It can be seen that the Kalman output error variances are significantly smaller than the measurement error variances. The Kalman filter is performing significant temporal filtering of the estimates resulting in improved estimation accuracy.

In order to verify the validity of this approach for general conditions, more simulation tests were performed with different object characteristics, different image noise level, different object motion velocities and different motion trajectories. The results showed that the Kalman filter provides much more accurate estimates when tracking a slower moving object. The Kalman filter provided more accurate estimates when image measurement errors were smaller, but large increase of measurement noise lead to relatively small increase in Kalman tracking errors. Selection of the $Q$ matrix affects the Kalman tracking accuracy. The weights in $Q$ should be selected larger when the relative motion is fast and selected smaller when the relative motion is slow. A constant $Q$ matrix does not significantly affect the Kalman tracking accuracy for different relative motion trajectories with similar motion velocities. The object feature locations and the number of features also affect the accuracy of Kalman estimates. Using well separated non-coplanar object features improves the Kalman filter estimates. This is because for coplanar object features close together, the corresponding image feature locations may not provide sufficient information for good pose parameter estimation. As expected, more object features result in more accurate Kalman estimates. However, five non-coplanar features were found to give good results and more than five did not result in significant improvement in tracking accuracy.

Real-time experiments were performed on the robot system to verify the simulation results and to demonstrate the feasibility of this approach when applied to a real system. The experiments were performed using the experimental system described in Section 3. In the experiments, the relative object motions were obtained by moving the end-effector mounted camera along defined spatial trajectories while the test object was fixed in the working space. The independent pose measure-

Table 2 Measurement Error Variance And Kalman Output Error Variance

| *features* | | #1 | #2 | #3 | #4 | #5 |
|---|---|---|---|---|---|---|
| *measurement* | $x$ | 0.06067 | 0.06035 | 0.06032 | 0.06063 | 0.06040 |
| *error variance* | $y$ | 0.06069 | 0.06047 | 0.06024 | 0.06039 | 0.06024 |
| *Kalman output* | $x$ | 0.010465 | 0.022139 | 0.021038 | 0.013151 | 0.017127 |
| *error variance* | $y$ | 0.017017 | 0.020883 | 0.012041 | 0.013940 | 0.012409 |

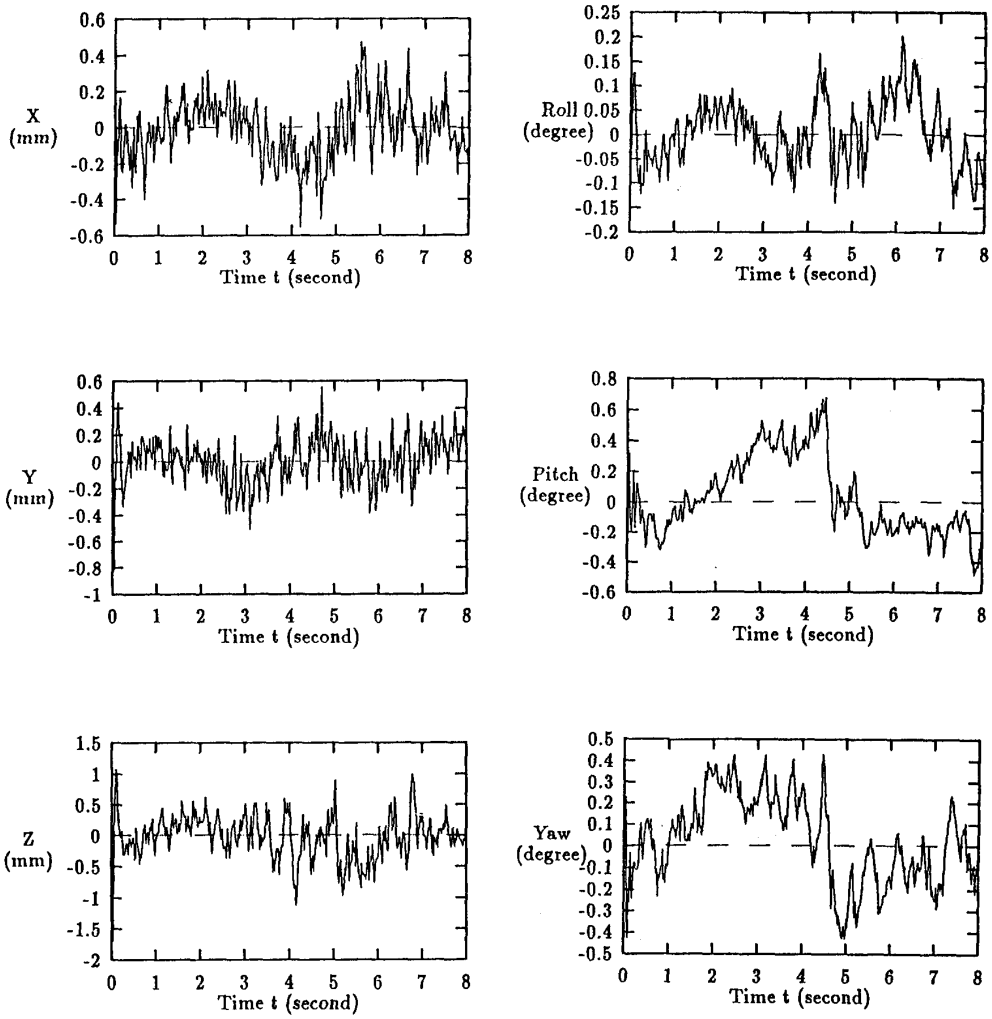


**Figure 3. Kalman Tracking Errors In Simulation**

Table 3 Output Error Variances

| *features* | | #1 | #2 | #3 | #4 | #5 |
|---|---|---|---|---|---|---|
| *forward kinematics* | $x$ | 0.43601 | 0.33497 | 0.36703 | 0.42007 | 0.37162 |
| *output error variance* | $y$ | 0.36069 | 0.31638 | 0.35061 | 0.31823 | 0.30387 |
| *Kalman output* | $x$ | 0.02587 | 0.02552 | 0.05711 | 0.01272 | 0.05611 |
| *error variance* | $y$ | 0.01668 | 0.03190 | 0.02426 | 0.02801 | 0.02114 |

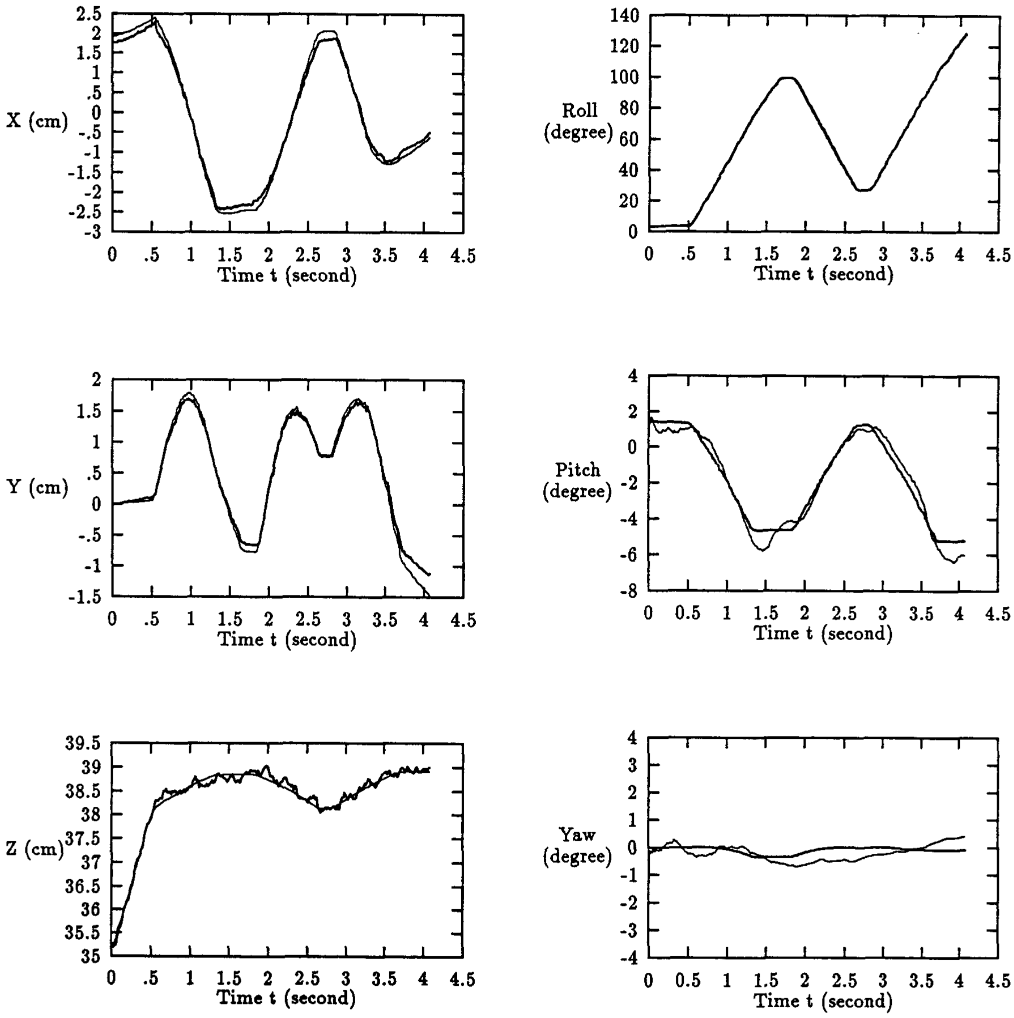


**Figure 4. Real-Time Motion Trajectory Estimates From Forward Kinematics And Kalman Filter**

ments obtained from the robot forward kinematics calculation were used to compare with the Kalman pose estimates. The image preprocessor used in the tests provides feature measurements with error variance approximately 0.06 $pixel^2$. The $R$ matrix in Kalman filter was selected to be same as it in the simulation. The disturbance noise covariance matrix $Q$ was determined through the system tests.

The experimental results substantiate the simulation results. The system provides more accurate pose estimates when relative motion is slower and non-coplanar features are used. Figure 4 shows real-time experimental results for 3D relative motion trajectories based on the robot forward kinematics calculations and Kalman filter estimates using five non-coplanar object features. It was found that the tracking agreement between the Kalman filter estimates and the robot forward kinematics solutions was within $\pm 1$ *mm* on $X$ and $Y$, within $\pm 2$ *mm* on $Z$, within $\pm 0.8$ *degree* on *Roll*, within $\pm 1.2$ *degree* on *Pitch*, and within $\pm 0.5$ *degree* on *Yaw*. However, it must be pointed out that the real tracking accuracy of the Kalman estimates is expected to be much better than those shown in Figure 4. This is because the robot forward kinematics solutions contain larger errors due to measurement errors of robot parameters and robot joint encoder data. Table 3 shows forward kinematics output error variances and Kalman output error variances. It can be seen that the forward kinematics solutions have much larger error variances than the Kalman filter solutions. Therefore, the Kalman filter solutions give much more confident estimates.

## 5 Conclusion

An approach for determining the 3D relative pose parameters of a random moving object with respect to a camera for 3D tracking control has been studied and analyzed in this paper. The extended Kalman filter is used to reduce the effect of vision measurement noise and to provide optimal pose estimates. Both computer simulation and real-time experimental results have demonstrated the effectiveness of this approach. Five non-coplanar object features are recommended for accurate estimation. The weights in dynamic disturbance noise covariance matrix $Q$ play an important role in the estimation. They should be selected small when relative motion is slow and large when relative motion is fast.

## References

[1] Red J. Broida and Rama Chellappa. Kinematics and structure of a rigid object from a sequence of noisy images. *IEEE Trans. Robotics Auto.*, page 95, 1986.

[2] Yuncai Liu, Thomas S. Huang and Olivier D. Faugeras. Determination of camera location from 2-d to 3-d line and point correspondences. *Trans. Pattern Ana. and Mach. Intel.*, page 28, 1990.

[3] Michel Dhome, Maarc Richetin, Jean-Thierry Lapreste and Gerard Rives. Determination of the attitude of 3-d objects from a single perspective view. *Trans. Pattern Ana. and Mach. Intel.*, page 1265, 1989.

[4] K. C. Chang, H. J. Lee and C. G. Chung. Adaptively estimating motion parameters for tracking a maneuvering target using image. *Inter. Robotics and Automation*, page 43, 1989.

[5] E. D. Dickmanns, B. Mysliwetz and T. Christians. An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles. *IEEE Trans. On Systems, Man, And Cybernetics*, page 1273, 1990.

[6] R. A. Smith. Master's thesis, Department of Electrical Engineering, University of Waterloo, 1988.

[7] S.T.Barnard. Choosing a basis for perceptual space. *Comput. Vision Graph. Image Processing*, page 87, 1985.

[8] Dave Westmore. Master's thesis, Department of Electrical Engineering, University of Waterloo, 1990.

[9] W.J. Wilson, editor. *Vision sensor integration for dynamic control of robotsi*, Detroit, Michigan, June 5-9, 1988. Robots 12 vision'88 Conference.

[10] Joseph S.-C Yuan. A general photogrammetric method for determining object position and orientation. *IEEE Trans. Robotics Auto.*, page 129, 1989.