

**Research on Intersample Prediction Control  
Schemes Based on Multirate GPC for  
High-Speed Visual Tracking Tasks**  
(高速ビジュアルサーボイングのためのマルチレート GPC に  
基づくサンプル点間予測制御)

**JiunDe Wu**

(呉 俊徳)



The Department of Electrical Engineering  
Graduate School of Engineering  
The University of Tokyo

A Thesis Submitted for  
the Degree of Master

**September 2003**

Supervisor :

**Professor Yoichi Hori**



**Research on Intersample Prediction Control  
Schemes Based on Multirate GPC for  
High-Speed Visual Tracking Tasks**

**JiunDe Wu**



**The Department of Electrical Engineering  
Graduate School of Engineering  
The University of Tokyo**

**A Thesis Submitted for  
the Degree of Master**

**September 2003**

**Supervisor :**

**Professor Yoichi Hori**



# Abstract

Visual servo system is a rapidly maturing approach to the control of robot manipulators that is based on visual recognition of robot and workpiece location. While robotics has thrived in certain environments, it has found challenges in environments that are not well defined. Visual servoing was developed mainly due to these challenges.

Machine vision can extend the feedback measurement space to include the relative position and orientation of the robot end-effector. The role of machine vision as the feedback transducer strongly affects the closed-loop dynamics of the overall system. Until now researches on visual servo system have focused mainly on preview control; only a few papers have focused on prediction control. However, all of them have not considered coordinate transformation problem caused by the multirate sampling characteristics of visual servo system while performing a high speed tracking task.

In this paper, two novel visual servo prediction control schemes are proposed for achieving high speed tracking and high control accuracy. In view of diagonalization problem and the coordinate transformation problem caused by low sampling frequency (long time-delay) in high speed target motion tracking, the intersample prediction control scheme based on fast rate GPC controller with an intersample predictor and the intersample prediction control scheme based on multirate GPC controller are proposed to take care of external uncertainties and compute the optimal intersample control inputs of the robotic system. Considering the effects of noise in prediction, simulations are performed to discuss the performance of noise suppression. Finally, simulation and experimental results are given to show the drastic effectiveness of the proposed approaches.



# Acknowledgments

First of all, I would like to express my sincere gratitude to my supervisor, Professor Yoichi Hori for giving me invaluable advice on this work. He was very kind to share his limited time to teach me not only how to be a researcher but also about Japanese culture. He always directed me to keep working with my research while enjoying it. I am very proud of belonging to his laboratory from 2001 to 2003, and these two years will be a precious treasure which I will carry through my life.

I would like to thank Matsushita Electric Industrial Co., Ltd. To receive Panasonic Scholarship is really an honor for me. The generous financial support allowed me to concentrate on my research without any worry. By holding seminars for us, Panasonic Scholarship Students, they helped us come into contact with well-known and well-applauded Japanese technology, culture and geography.

I am greatly thankful to the previous and current members of my laboratory. Mr. Uchida, the technical staff, helped me both in the technical and official works to let me concentrate on this research. Discussion with my colleagues, Dr. Oyobe, Seki, Oh, Ma, Hata, Bando and all members of my laboratory, always stimulated me. They always shared their research experience to me and gave me suggestions. My heartfelt thanks to Mr. Okano, Tai, Chang and Ms. Inoue who made my stay in Japan and Hori laboratory an unforgettable experience.

Finally, I would like to thank all my family members and good friends for their encouragement and support.

JiunDe Wu  
The University of Tokyo  
August 2003



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background of the Research . . . . .	1
1.2	Motivations of the Thesis . . . . .	2
1.3	Outline of the Thesis . . . . .	4
<b>2</b>	<b>Fundamentals of Visual Servoing</b>	<b>7</b>
2.1	Abstract . . . . .	7
2.2	Image-Based versus Position-Based Visual Servo Control . . . . .	7
2.3	Machine Vision . . . . .	8
2.3.1	Perspective Model . . . . .	8
2.3.2	Calibration of Visual Sensor . . . . .	9
2.4	Position-Based Visual Servoing . . . . .	11
2.4.1	Computing Mapping matrix and Kinematic Jacobian . . . . .	11
2.4.2	Deriving Control Law . . . . .	12
2.5	A Two-Link Manipulator Case . . . . .	12
2.6	Summary . . . . .	15
<b>3</b>	<b>Modeling the Visual Servo Loop</b>	<b>17</b>
3.1	Abstract . . . . .	17
3.2	Linearized Model of Joint Servo System . . . . .	17
3.3	Multirate Sampling Characteristics of Visual Servo System . . . . .	19
3.4	Diagonalization Problem and Coordinate Transformation Problem . . . . .	21
3.4.1	Conventional Visual Servoings . . . . .	22
3.4.2	Alternative Visual Servoing . . . . .	24
3.5	Summary . . . . .	24
<b>4</b>	<b>Intersample Prediction Control Schemes Based on Multirate GPC for High-Speed Tracking Tasks</b>	<b>27</b>
4.1	Abstract . . . . .	27

4.2	The Concepts of Proposed Approaches . . . . .	27
4.2.1	Fastrate controller with Intersample Predictor . . . . .	28
4.2.2	Multirate controller with Estimator . . . . .	28
4.3	Target State Estimation . . . . .	29
4.4	Lifting GPC Controller . . . . .	31
4.5	Summary . . . . .	37
<b>5</b>	<b>Verifications of the Effects on Proposed Schemes</b>	<b>39</b>
5.1	Abstract . . . . .	39
5.2	Simulation Verifications . . . . .	39
5.2.1	Circular Target Motion Tracking without noise . . . . .	40
5.2.2	Square Target Motion Tracking without noise . . . . .	41
5.2.3	Circular Target Motion Tracking with noise . . . . .	41
5.2.4	Square Target Motion Tracking with noise . . . . .	43
5.3	Experimental Verifications . . . . .	43
5.3.1	Experimental Hardware and Software . . . . .	44
5.3.2	Circular Target Motion Tracking . . . . .	48
5.3.3	Square Target Motion Tracking . . . . .	49
5.4	Summary . . . . .	51
<b>6</b>	<b>Conclusion</b>	<b>53</b>
6.1	Summary . . . . .	53
6.2	Problems and Future Works . . . . .	54
6.2.1	Timing Problem of RTLinux . . . . .	54
6.2.2	High-Ordered Predictive Model . . . . .	55
<b>A</b>	<b>The Experimental Setups of Machine Vision</b>	<b>59</b>
<b>B</b>	<b>The Software for Visual Servoing</b>	<b>61</b>

# List of Figures

1.1	The RTS PixCell <sup>TM</sup> concept . . . . .	2
1.2	Robot ping-pong player . . . . .	2
1.3	MIT's Intelligent Transportation Research . . . . .	2
1.4	Visual feedback control system . . . . .	3
1.5	Visual servo with feedback and feedforward compensation . . . . .	4
1.6	Visual servo with feedback and estimated feedforward compensation . . . . .	4
2.1	Image-based visual servo control . . . . .	8
2.2	Position-based visual servo control . . . . .	8
2.3	Perspective model . . . . .	9
2.4	Camera calibration flow chart . . . . .	10
2.5	Image distortion . . . . .	10
2.6	Position-based visual servo structure . . . . .	13
2.7	Two-link manipulator . . . . .	13
3.1	Joint servo system . . . . .	18
3.2	Linearized dynamical model of the visual servo system . . . . .	18
3.3	Multirate sampling visual servo system . . . . .	20
3.4	(a)Single-rate (b)Multirate sampling control . . . . .	20
3.5	Low speed tracking $\pi/3[rad/s]$ . . . . .	21
3.6	High speed tracking $1\pi[rad/s]$ . . . . .	21
3.7	Low speed tracking $200[mm/s]$ . . . . .	21
3.8	High speed tracking $400[mm/s]$ . . . . .	21
3.9	Workspace servo system without updating angle . . . . .	23
3.10	Workspace servo system with updating angle . . . . .	23
3.11	Workspace servo system with updating angle and prediction . . . . .	24
3.12	Workspace servo system with updating angle and intersample prediction . . . . .	24
4.1	Conventional visual servo control scheme . . . . .	28
4.2	Conventional visual servo control scheme with prediction . . . . .	28

4.3	Proposed visual servo control scheme-fastrate GPC controller with inter-sample predictor . . . . .	29
4.4	Proposed visual servo control scheme-taking multirate GPC controller as intersample predictor . . . . .	29
4.5	Block diagram of Kalman Filter . . . . .	30
4.6	Basic structure of Kalman Filter . . . . .	30
4.7	Block diagram of GPC . . . . .	31
4.8	Basic structure of single-rate GPC . . . . .	32
4.9	Basic algorithm of single-rate GPC . . . . .	32
4.10	Basic structure of fastrate GPC . . . . .	34
4.11	Basic algorithm of fastrate GPC . . . . .	34
4.12	Basic structure of multirate GPC . . . . .	36
4.13	Basic algorithm of multirate GPC . . . . .	36
5.1	Tracking trajectory for circular target motion(simulation without noise) . .	40
5.2	Tracking response for circular target motion(simulation without noise) . . .	41
5.3	Tracking error for circular target motion(simulation without noise) . . . .	41
5.4	Tracking trajectory for square target motion(simulation without noise) . .	42
5.5	Tracking response for square target motion(simulation without noise) . . .	42
5.6	Tracking error for square target motion(simulation without noise) . . . .	42
5.7	Tracking trajectory for circular target motion in proposed approaches(simulation with noise) . . . . .	43
5.8	Tracking trajectory for square target motion in proposed approaches(simulation with noise) . . . . .	43
5.9	Tracking trajectory for circular target motion(simulation with noise) . . . .	44
5.10	Tracking response for circular target motion(simulation with noise) . . . .	44
5.11	Tracking error for circular target motion(simulation with noise) . . . . .	45
5.12	Tracking trajectory for square target motion(simulation with noise) . . . .	45
5.13	Tracking response for square target motion(simulation with noise) . . . .	46
5.14	Tracking error for square target motion(simulation with noise) . . . . .	46
5.15	Experimental setup . . . . .	47
5.16	Photographs of 2-link DD robot with webcam . . . . .	47
5.17	Circular motion target . . . . .	47
5.18	Structure of square motion target . . . . .	48
5.19	Experimental Software . . . . .	48
5.20	Tracking trajectory for circular target motion(experiment) . . . . .	49
5.21	Tracking response for circular target motion(experiment) . . . . .	49

5.22	Tracking error for circular target motion(experiment) . . . . .	50
5.23	Tracking trajectory for square target motion(experiment) . . . . .	50
5.24	Tracking response for square target motion(experiment) . . . . .	51
5.25	Tracking error for square target motion(experiment) . . . . .	51
6.1	Details of single-rate system timing . . . . .	54
6.2	Details of multirate system timing . . . . .	54
6.3	Predictive model incorporated a phase-compensated parameter . . . . .	56
6.4	Prediction error while tuning K and $\alpha$ . . . . .	57
B.1	Real time task create, FIFO interlinked diagram . . . . .	61
B.2	Camera action flow chart . . . . .	62

# List of Tables

A.1 Softwares used for visual servoing . . . . . 59  
A.2 Elements of machine vision . . . . . 59

# Notation

$\triangleq$	defined as
$\mathbf{A}$	matrix
$\mathbf{x}$	row or column vector
$\mathbf{I}$	identity matrix
$\mathbf{0}$	zero matrix
$\mathbf{A}^T$	transpose of $\mathbf{A}$
$\mathbf{A}^{-1}$	inverse of $\mathbf{A}$
$G(s)$	continuous-time system
$G(z)$	discrete-time system
$x$	continuous-time signal
$x_k$	discrete-time signal
$x_{\nu_k}$	lifted discrete-time signal
$\ \cdot\ $	norm of vector or matrix
$\ \mathbf{x}\ _2$	$\triangleq \sqrt{\sum_{i=1}^n  x_i ^2}$ : the 2-norm of vector $\mathbf{x}$
$Z[\cdot]$	$z$ -transformation



# Chapter 1

## Introduction

### 1.1 Background of the Research

Although human beings have five senses(touch, taste, hearing, eyesight, and smell), more than 87% external information is obtained by eyes. Thus, it has no doubt that eyes are the most important sensory organs for obtaining intellectual information. In future direction of robotics, we can expect that visual servoing will play an important role.

Since 20 years ago, visual servo control started to be applied in robotics. In an automatic manufacturing system incorporated with robots, all the work operations are taken repeatedly in equipped environment. In this case, traditional robot control based only on internal sensor, i.e. the angle encoder and tachometer, is sufficient to take high accurate control. However performing autonomous work operation in uncertain environment is impossible.

Machine vision gives robots the ability to manipulate parts with uncertain characteristics and locations. One of the most important applications is vision-based material handling[1](shown in Fig. 1.1). In entertainment, the construction of robot player[16](shown in Fig. 1.2) to play ping-pong against humans and machines is quite an interesting challenge. Besides, in ITS(Intelligent Transportation System)[2](shown in Fig. 1.3), a monitoring system which can recognize road lanes from vision is necessary. As a consequence, no matter what kind of target motion, i.e. linear, circular, square or sinusoidal trajectory, it must be able to be estimated and be predicted with sufficient accuracy. Realizing a high performance control system of tracking a high speed mobile target with the help of dynamic visual feedback is the goal for all the applications in visual servo control system. Thus we are confronted with a control problem of tracking a mobile target with the help of dynamic visual feedback.

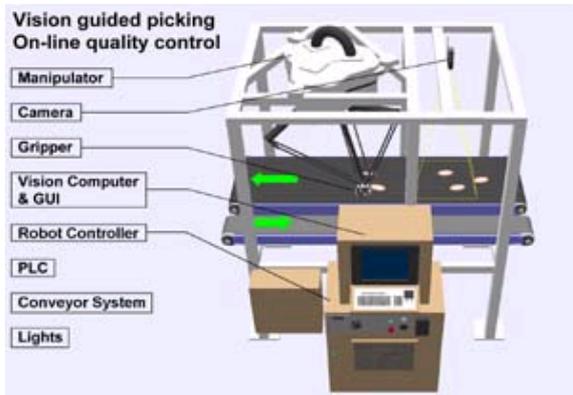


Figure 1.1: The RTS PixCell<sup>TM</sup> concept



Figure 1.2: Robot ping-pong player

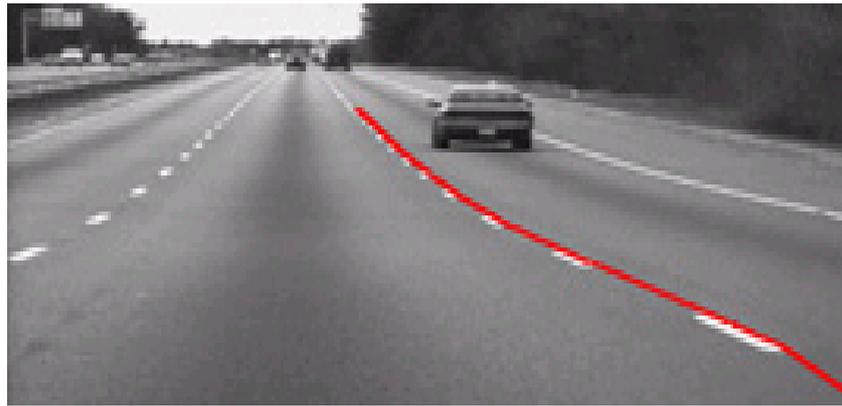


Figure 1.3: MIT's Intelligent Transportation Research

## 1.2 Motivations of the Thesis

Some control strategies have been previously researched. Corke et al.[3][35][36] were the first who studied the effects of the manipulator's dynamics in the visual loop. Nakabo et al.[14] used a  $1kHz$  sampling rate vision chip for visual servoing. Nam et al.[15] used PSD sensor to perform high performance visual tracking. Hashimoto et al.[4][21] proposed observer-based controller to overcome delay and nonlinear dynamics. Gangloff et al.[5][6] were the first that implemented GPC online to take into account the dynamics of the manipulator and verified that GPC controller has higher control performance than PI controller. In recent years, multirate sampling control theories have been extensively

studied[12][13][19][20]. Based on it, some multirate visual servoing approaches[11][26][22] were proposed.

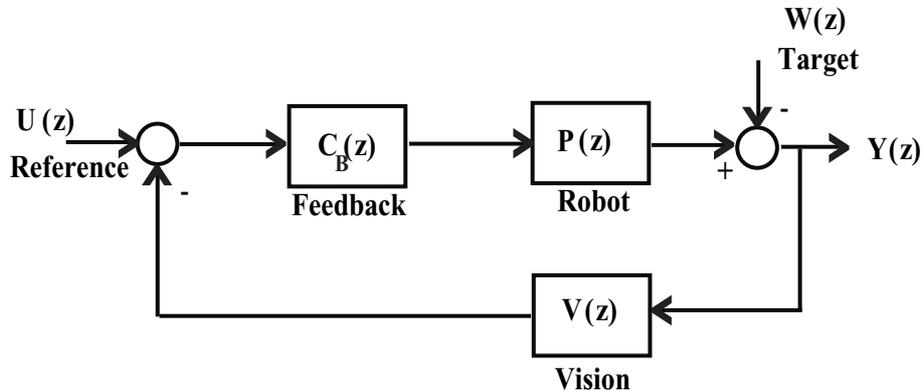


Figure 1.4: Visual feedback control system

In feedback control (shown in Fig. 1.4), machine vision has a number of significant disadvantages when used as a feedback sensor: a relatively low sampling rate, sensitive to variations of the environment illumination and coarse quantization. As a result, direct use of the visual data for robot control will lead to poor control accuracy.

To reduce these effects, some recent researches on visual servo system have focused mainly on using a high-price machine vision system to obtain a high sampling rate, high resolution machine vision system. Instead of using high performance vision system, some other researches took a low speed motion control or small area motion control so the robot only moves a little during sampling period. Although there are many researches in visual servo control[23][28][31][32][33][34][38], most researches design visual servoing for target tracking in equipped environment so that preview control theory (shown in Fig. 1.5) can be used to perform a high accuracy and high speed tracking work. But the merit of machine vision applied in robotics is to make robots be able to operate autonomously in uncertain environment is neglected.

In this paper, in order to focus on operating autonomous work, prediction control (shown in Fig. 1.6) is considered here instead of preview control. A well-known characteristics in prediction method is that the higher dynamic prediction performance the system has, the lower is its stochastic performance. So the effects of noise in prediction is also considered in this research. While performing high speed tracking in visual servo control, there exists the coordinate transformation problem caused by multirate characteristic that is still not considered. To overcome this problem, instead of using a high-priced machine vision system, a web camera was used. Furthermore, intersample predictor control schemes based on multirate GPC[13] for high-speed visual tracking tasks were proposed to solve low

sampling rate problem and improve the control accuracy in tracking trajectory.

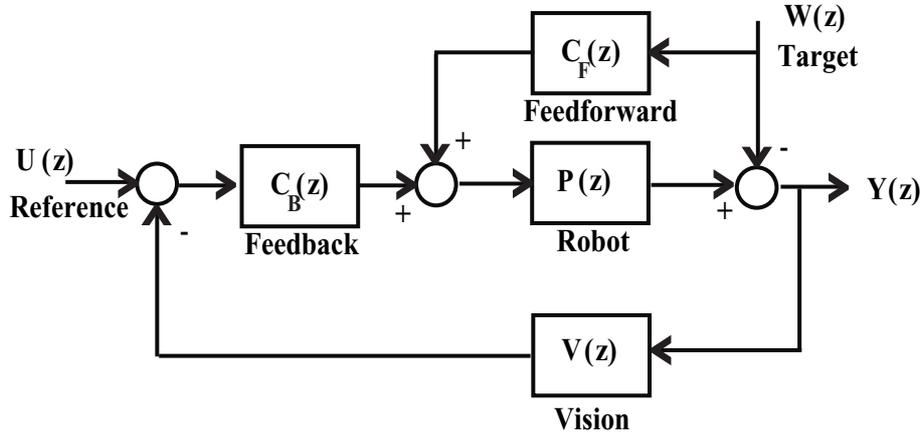


Figure 1.5: Visual servo with feedback and feedforward compensation

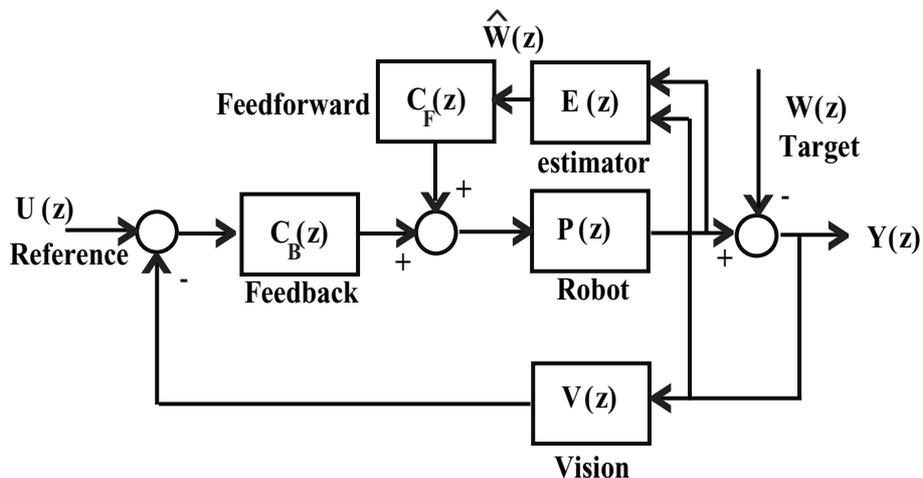


Figure 1.6: Visual servo with feedback and estimated feedforward compensation

### 1.3 Outline of the Thesis

In chapter 1, the background of this research and the motivations of this thesis were described. In chapter 2, fundamentals of visual servoing about machine vision and position-based visual servoing are described. In chapter 3, the method to model the visual servo loop is explained. The diagonalization problem and the coordinate transformation problem caused by the multirate characteristics of visual servo system are explained. Simulation results are given to show the effects of these two problems. Chapter 4 is the discussion

of proposed novel visual servo control approaches, intersample prediction control schemes based on multirate GPC for high-speed visual tracking tasks. In chapter 5, the simulation and experimental results of circular target motion tracking task and square target motion tracking task are given to show the drastic performance of our proposals. Finally, a conclusion summarises the contributions made by this thesis and outlines directions for future research.



# Chapter 2

## Fundamentals of Visual Servoing

### 2.1 Abstract

Visual servoing is the fusion of results from many elemental areas including high-speed image processing, kinematics, dynamics, control theory, and real-time computing. The two major classes of systems, image-based visual servo systems and position-based visual servo systems are discussed in Section 2.2. The reason why we choose position-based control is also explained in this section. Since feature extraction occupies an important position in visual servoing, Section 2.3 describes perspective model and the calibration approach to obtain precise image features. Section 2.4 presents the control law of position-based visual servoing. Finally a two-link manipulator case is derived in Section 2.5.

### 2.2 Image-Based versus Position-Based Visual Servo Control

Image-based control(see Fig. 2.1) and position-based control(see Fig. 2.2) are the two most classical categories of visual servoing[3][29] and they are different in the nature of the inputs used in their respective control schemes. Image-based visual servoing is well known to be generally satisfactory, even in the presence of important camera calibration errors[8]. It acts on an error signal which is defined by image features, such that when the servoing goal is reached, the error is zero. The error may be calculated by comparison with a pre-recorded image, i.e. a teach-by-showing approach. However in industrial settings, the desired target location is likely to be unknown or mobile. In addition, there are some stability and convergence problems as investigated by Chaumette et al[9]. Therefore, a position-based approach is used in this research. A eye-in-hand camera configuration is

chosen in our work since we are servoing mostly to a single point, bringing it to the center where there is less distortion. Another merit of this configuration is that the feature extraction is assumed to be simple in order to focus more on the control problem of the visual servo system.

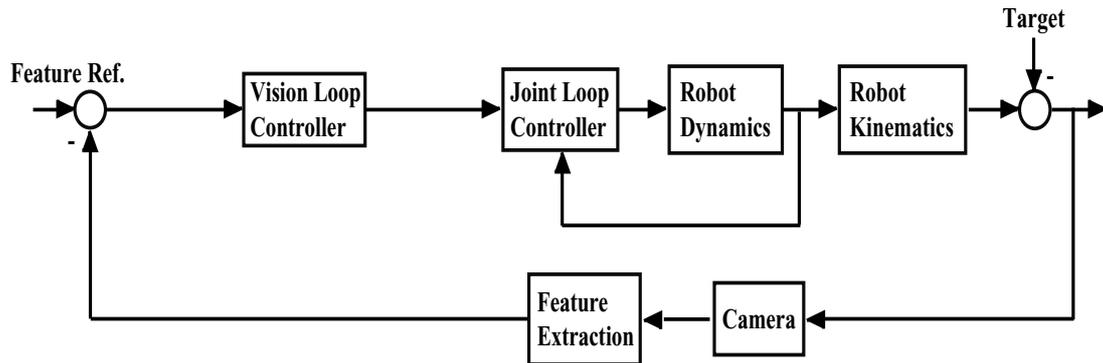


Figure 2.1: Image-based visual servo control

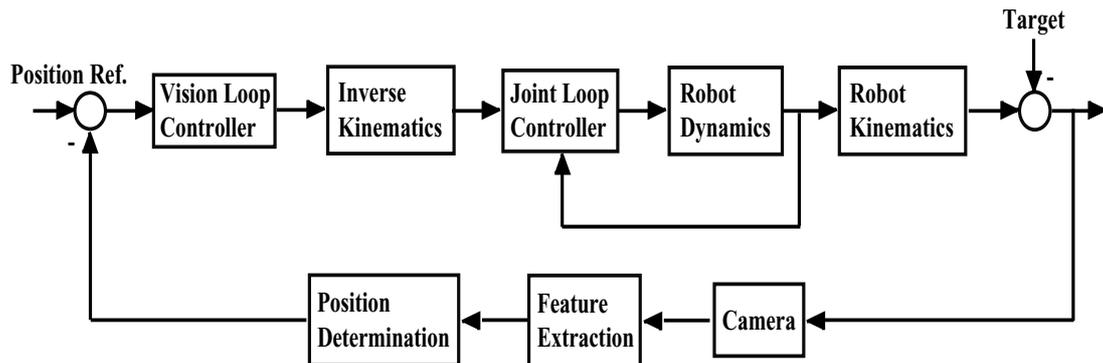


Figure 2.2: Position-based visual servo control

## 2.3 Machine Vision

### 2.3.1 Perspective Model

Suppose that the camera is mounted on the end-effector of the robot. In Fig. 2.3,  $\mathbf{p} \triangleq (x, y, z)$  on the camera coordinate system is determined only by the relative position between the camera position and object position. Therefore, the perspective model of the

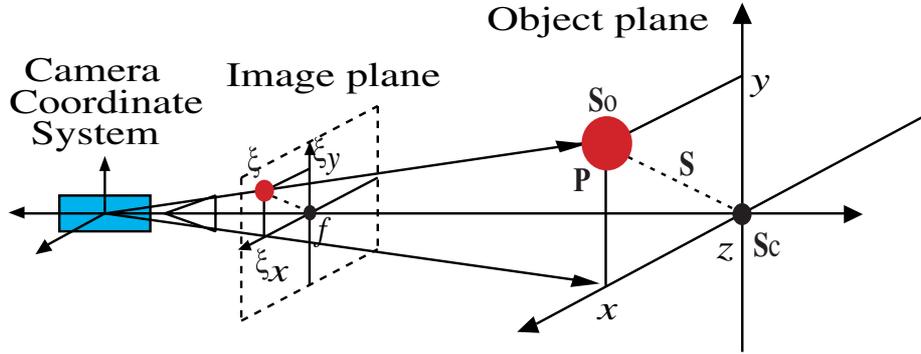


Figure 2.3: Perspective model

camera can be obtained because  $\mathbf{p}$  is mapped to the feature point  $\xi$  on the image plane.

$$\xi = \begin{bmatrix} \xi_x \\ \xi_y \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{z} (\mathbf{s}_c - \mathbf{s}_o) \quad (2.1)$$

Here,  $f$  is the focus distance,  $z$  is the distance between the object and camera in the  $Z$  axis direction and  $\theta$  is the joint variable vector.

### 2.3.2 Calibration of Visual Sensor

The projective geometry of the camera on the end-effector is modeled by a perspective projection. Further, the distortion introduced by the lens is also added to this model. Thus,  $\mathbf{p} = [x \ y \ z]^T$  which are the coordinates of a point with respect to the camera coordinate system, will project onto the image plane with coordinates  $\xi = (X_f, Y_f)$  in pixels. Here we take Tsai's approach[10](see Fig. 2.4) to solve for the camera's transformation and distortion.

#### Viewport Calculation:

Calculate actual discrete pixel address location,  $(X_f, Y_f)$  within image, with given centering

$$\begin{bmatrix} \xi_x \\ \xi_y \\ 1 \end{bmatrix} = \begin{bmatrix} X_f \\ Y_f \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & C_x \\ 0 & S_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_d \\ Y_d \\ 1 \end{bmatrix} \quad (2.2)$$

When the camera is mounted on the end-effector,  $C_x = C_y = 0$ .  $S_x$  and  $S_y$  are the magnification factors for the  $x$  and  $y$  directions, respectively because the dimensions

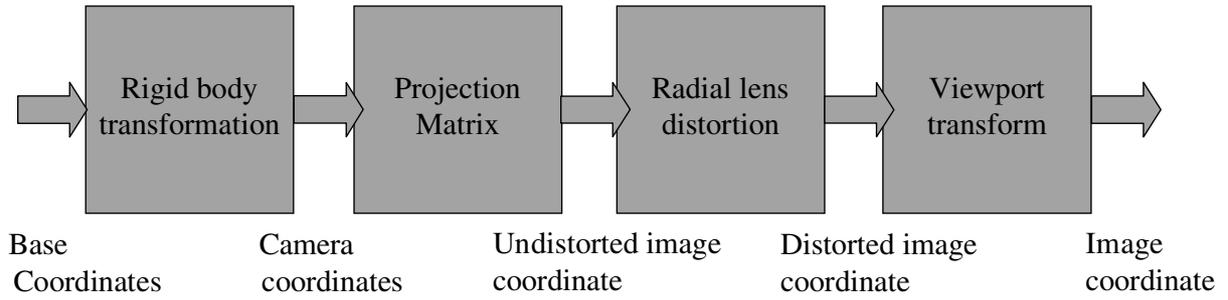


Figure 2.4: Camera calibration flow chart

of the pixels in vision chip are not square. We can obtain these information from the specifications of camera.  $(X_d, Y_d)$  is the distorted or true image coordinate on the image plane.

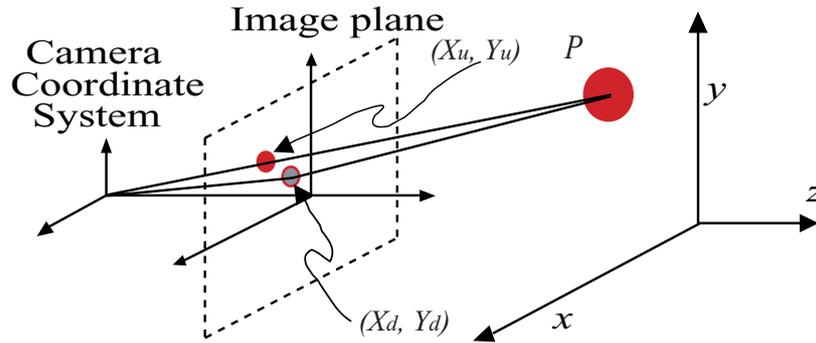


Figure 2.5: Image distortion

### Radial Distortion:

This distortion is caused because the focus of the lens is not always the same in any part of the lens(See Fig. 2.5).  $(X_u, Y_u)$  is the undistorted coordinate. The distortion is calculated by

$$\begin{bmatrix} X_d \\ Y_d \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & D_x \\ 0 & 1 & D_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \\ 1 \end{bmatrix} = \frac{1}{D} \begin{bmatrix} X_u \\ Y_u \\ 1 \end{bmatrix} \quad (2.3)$$

where  $D_x = X_d(K_d r^2)$ ,  $D_y = Y_d(K_d r^2)$ ,  $D = 1 - K_d r^2$ ,  $r = \sqrt{X_d^2 + Y_d^2}$ ,  $K_d$  is the distortion coefficient.

### Perspective Projection:

Camera coordinates to ideal undistorted image coordinate, involves focal length,  $f$ , and perspective foreshortening

$$\begin{bmatrix} X_u \\ Y_u \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{z} & 0 & 0 \\ 0 & \frac{f}{z} & 0 \\ 0 & 0 & \frac{1}{z} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.4)$$

From (2.4)(2.3)(2.2), the relational equation from camera coordinate to actual pixel address location within the image is given by

$$\begin{bmatrix} X_f \\ Y_f \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{G_x}{z} & 0 & 0 \\ 0 & \frac{G_y}{z} & 0 \\ 0 & 0 & \frac{1}{z} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{G} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.5)$$

where  $G_x = \frac{S_x f}{D}$ ,  $G_y = \frac{S_y f}{D}$  and  $\mathbf{G}$  is calibration matrix.

## 2.4 Position-Based Visual Servoing

### 2.4.1 Computing Mapping matrix and Kinematic Jacobian

Let  $\theta$  be the current joints' position vector. And let  $\mathbf{M}_{bc}(\theta)$  be the homogeneous transformation between the base coordinate system of the robot and camera coordinate system.

$$\begin{bmatrix} \mathbf{s}_o \\ 1 \end{bmatrix} = \mathbf{M}_{bc}(\theta) \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}, \quad \mathbf{M}_{bc}(\theta) = \begin{bmatrix} \mathbf{R}_{bc}(\theta) & \mathbf{T}_{bc}(\theta) \\ \mathbf{0} & 1 \end{bmatrix} \quad (2.6)$$

where  $\mathbf{R}_{bc}$  is a rotation matrix and  $\mathbf{T}_{bc}$  is a translation vector. We know  $\mathbf{s}_o = \mathbf{s}_c + \mathbf{s}$ . From (2.6), the following equations are obtained

$$\mathbf{s} = \mathbf{R}_{bc}(\theta)\mathbf{p} = \mathbf{R}_{bc}(\theta)\mathbf{G}^{-1}\xi = \iota^{-1}(\theta)\xi \quad (2.7)$$

$$\mathbf{s}_c = \mathbf{T}_{bc}(\theta) \quad (2.8)$$

where  $\iota(\theta)$  is the base coordinate system to camera coordinate system mapping matrix. In servo control, control input should be transformed to angle coordinate system. Now, let's define  $\mathbf{J}$  as the kinematic Jacobian linking  $\dot{\mathbf{s}}_c$ , the velocity screw of the camera, to  $\dot{\theta}$  the velocity of the joint coordinate

$$\dot{\mathbf{s}}_c = \mathbf{J}(\theta)\dot{\theta} \quad (2.9)$$

with  $\dot{\theta} = [\dot{\theta}_1, \dots, \dot{\theta}_n]$ . The kinematic Jacobian  $\mathbf{J}(\theta)$  can be derived from the following set of equations

$$\mathbf{J}(\theta) = \begin{bmatrix} \frac{\partial \mathbf{s}_c}{\partial \theta_1} & \dots & \frac{\partial \mathbf{s}_c}{\partial \theta_n} \end{bmatrix} \quad (2.10)$$

The inverse kinematic Jacobian  $\mathbf{J}^{-1}(\theta)$  can transform the control input from workspace to joint space by the following equation

$$\dot{\theta} = \mathbf{J}^{-1}(\theta)\dot{\mathbf{s}}_c \quad (2.11)$$

## 2.4.2 Deriving Control Law

The relative position,  $\mathbf{s}$ , is the difference between the camera position,  $\mathbf{s}_c$ , and target position  $\mathbf{s}_t$ . The control law can be obtained by differentiating  $\mathbf{s}$

$$\dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \theta} \dot{\theta} + \frac{\partial \mathbf{s}}{\partial \mathbf{p}} \dot{\mathbf{p}} \triangleq \mathbf{J}(\theta)\dot{\theta} + \mathbf{L}(\mathbf{p})\dot{\mathbf{p}} \quad (2.12)$$

where  $\mathbf{J}(\theta)$  is the Kinematic Jacobian shown in (2.10),  $\mathbf{L}(\mathbf{p})$  is the Target Motion Jacobian.

Discretizing equation(2.12), we can obtain the following equation

$$\Delta \mathbf{s}_k = \mathbf{J}(\theta_k)\Delta \theta_k + \mathbf{L}(\mathbf{p}_k)\Delta \mathbf{p}_k \quad (2.13)$$

Because  $\Delta \theta$  is independent from  $\Delta \mathbf{p}$  and  $\mathbf{p}$  is uncontrollable, the position error  $\Delta \mathbf{s}_k$  should only be compensated by moving the manipulator as shown in the following equation:

$$\Delta \theta_k = \mathbf{J}^{-1}(\theta_k)\Delta \mathbf{s}_k \quad (2.14)$$

Using equation(2.14), we can construct the position based visual servo system as shown in Fig. 2.6.

## 2.5 A Two-Link Manipulator Case

In this research, the experimental equipment is a 2-link DD robot. So an illustrative example is described here. A 2-link eye-in-hand manipulator can move only in planar surface, motion in  $z$  direction can be neglected. Thus, we can simplify the calculation by omitting the  $z$  element. First, let us calculate the homogeneous transformation matrix  $\mathbf{M}_{bc}$  between the base frame of robot and the camera frame. From Fig. 2.7 the homogeneous coordinate transformation matrix is obtained by



Thus  $\mathbf{R}_{bc}(\theta)$  and  $\mathbf{T}_{bc}(\theta)$  are

$$\mathbf{R}_{bc}(\theta) = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (2.18)$$

$$\mathbf{T}_{bc}(\theta) = \begin{bmatrix} L_2 \cos(\theta_1 + \theta_2) + L_1 \cos \theta_1 \\ L_2 \sin(\theta_1 + \theta_2) + L_1 \sin \theta_1 \end{bmatrix} \quad (2.19)$$

Then the dynamic position of camera is  $\mathbf{s}_c = T_{bc}(\theta)$ .

We can also simplify the calibration matrix from (2.5) by neglecting the  $z$  axis.

$$\mathbf{G} = \begin{bmatrix} \frac{G_x}{z} & 0 \\ 0 & \frac{G_y}{z} \end{bmatrix} \quad (2.20)$$

From (2.7), (2.18) and (2.20), the mapping matrix  $\iota(\theta)$  and the inverse mapping matrix  $\iota^{-1}(\theta)$  are given by

$$\begin{aligned} \iota(\theta) &= \mathbf{G}\mathbf{R}_{bc}^{-1}(\theta) \\ &= \frac{1}{z} \begin{bmatrix} G_x \cos(\theta_1 + \theta_2) & G_x \sin(\theta_1 + \theta_2) \\ -G_y \sin(\theta_1 + \theta_2) & G_y \cos(\theta_1 + \theta_2) \end{bmatrix} \end{aligned} \quad (2.21)$$

$$\begin{aligned} \iota^{-1}(\theta) &= \mathbf{R}_{bc}(\theta)\mathbf{G}^{-1} \\ &= \frac{z}{G_x G_y} \begin{bmatrix} G_y \cos(\theta_1 + \theta_2) & -G_x \sin(\theta_1 + \theta_2) \\ G_y \sin(\theta_1 + \theta_2) & G_x \cos(\theta_1 + \theta_2) \end{bmatrix} \end{aligned} \quad (2.22)$$

From (2.8),(2.10) and (2.19), the kinematic Jacobian  $\mathbf{J}(\theta)$  and inverse kinematic Jacobian  $\mathbf{J}^{-1}(\theta)$  are given by

$$\begin{aligned} \mathbf{J}(\theta) &= \begin{bmatrix} \frac{\partial \mathbf{s}_c}{\partial \theta_1} & \frac{\partial \mathbf{s}_c}{\partial \theta_2} \end{bmatrix} \\ &= \begin{bmatrix} -L_2 \sin(\theta_1 + \theta_2) - L_1 \sin \theta_1 & -L_2 \sin(\theta_1 + \theta_2) \\ L_2 \cos(\theta_1 + \theta_2) + L_1 \cos \theta_1 & L_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \end{aligned} \quad (2.23)$$

$$\begin{aligned} \mathbf{J}^{-1}(\theta) &= \frac{1}{L_1 L_2 \sin \theta_2} \\ &\times \begin{bmatrix} L_2 \cos(\theta_1 + \theta_2) & L_2 \sin(\theta_1 + \theta_2) \\ -L_2 \cos(\theta_1 + \theta_2) - L_1 \cos \theta_1 & -L_2 \sin(\theta_1 + \theta_2) - L_1 \sin \theta_1 \end{bmatrix} \end{aligned} \quad (2.24)$$

## 2.6 Summary

Three basic areas, image processing, kinematics and control theory were described. The reason why we chose position-based control as our control system was explained. The general problem in position-based control was its difficulty to perform position determination due to image distortion. The perspective model and the calibration approach we used to conquer this problem were described. A brief derivation of the position-based control law was explained. Finally, an illustrative example based on 2-link DD robot was described.



# Chapter 3

## Modeling the Visual Servo Loop

### 3.1 Abstract

Before designing a controller, deriving an open-loop transfer function from the linearized dynamical model of the visual servo system shown in Fig. 3.2 is necessary. By definition, position-based visual servoing requires a precise camera calibration model and a precise knowledge of the kinematic robot model. Errors in these models can result in large positioning errors. Thus in modeling the vision loop, we should take these two problems into consideration. The former problem is solved by Tsai's approach mentioned in Chapter 2. Section 3.2 describes the approach used to solve the latter problem. Since the effects of multirate sampling are remarkable in visual servo system, the multirate sampling characteristics of visual servoing is introduced in Section 3.3. A diagonalization problem occurs due to a long latency. Although modern researches solved it by utilizing multirate characteristics but coordinate transformation problem occurs. Section 3.4 discusses these two problems.

### 3.2 Linearized Model of Joint Servo System

In modeling the robot dynamics, we linearize this model by considering that the nonlinear effects act like slow load disturbances on the joint servo loop. The assumption is validated through the linear identification of an industrial manipulator[30][37][40]. The joint servo velocity controller is designed in order to control the robot as shown in Fig. 3.1. Because this controller employs the robust disturbance observer(DOB)[24] that is constructed in the joint space, the total stability of motion controller is independent of the configuration of robot, so each joint is decoupled. Therefore, the transfer function from the acceleration command  $\ddot{\theta}^{ref}$  to the robot velocity  $\dot{\theta}$  can be regarded as an integrator



$$\mathbf{J}^{-1}(\theta_{k-1}) = \begin{bmatrix} \mathbf{J}_{11}^{-1}(\theta_{k-1}) & \cdots & \mathbf{J}_{1n}^{-1}(\theta_{k-1}) \\ \vdots & \ddots & \vdots \\ \mathbf{J}_{n1}^{-1}(\theta_{k-1}) & \cdots & \mathbf{J}_{nn}^{-1}(\theta_{k-1}) \end{bmatrix} \quad (3.3)$$

Moreover, the discrete open loop transfer function  $\mathbf{G}(z)$  and workspace transfer function  $\mathbf{H}(z)$  can be written as

$$\mathbf{G}(z) = \frac{\mathbf{I}}{z} \frac{z\mathbf{I}}{z-1} \mathbf{H}(z) \quad (3.4)$$

$$\mathbf{H}(z) = \mathbf{J}(\theta_k) \mathbf{P}(z) \mathbf{J}^{-1}(\theta_{k-1}) \quad (3.5)$$

where

$$\mathbf{H}(z) = \begin{bmatrix} D_{11}(z) & C_{12}(z) & \cdots & C_{1n}(z) \\ C_{21}(z) & D_{22}(z) & \cdots & C_{2n}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1}(z) & C_{n2}(z) & \cdots & D_{nn}(z) \end{bmatrix} \quad (3.6)$$

$$C_{ik}(z) = \sum_{l=1}^n J_{il} P_l(z) J_{il}^{-1} \quad l = 1, \dots, n \quad (3.7)$$

$$D_{ii}(z) = \sum_{l=1}^n J_{il} P_l(z) J_{il}^{-1} \quad l = 1, \dots, n \quad (3.8)$$

Since the DC gain of  $\mathbf{G}(z)$  is the identity matrix, n degree-of-freedom robot(MIMO) can be considered in practice as n decoupled systems by neglecting the cross terms of  $\mathbf{G}(z)$ . Using open-loop transfer function  $\mathbf{G}(z)$ , we can design a controller for the visual servo loop.

### 3.3 Multirate Sampling Characteristics of Visual Servo System

A conventional multirate sampling visual servo control diagram is shown in Fig. 3.3. Two feedback loops are in this system. One is the visual servo loop and another one is the joint servo loop. However, the vision servo loop sampling period,  $T_f$ , of vision sensor such as a CCD camera is comparatively slow(over 33ms). The joint servo loop sampling period,  $T_j$ , of inner sensor such as the angle encoder is quiet fast(less than 1ms). The control input period  $T_u$  is generally decided by the speed of the actuator, D/A converter, or the calculation on the CPU.

Although there exists many time periods in visual control system, the conventional visual servo control make all periods equal to the longest one,  $T_f$ , for simplification(shown in Fig. 3.4(a)) because the rates that are not integrally related are difficult to analyze.

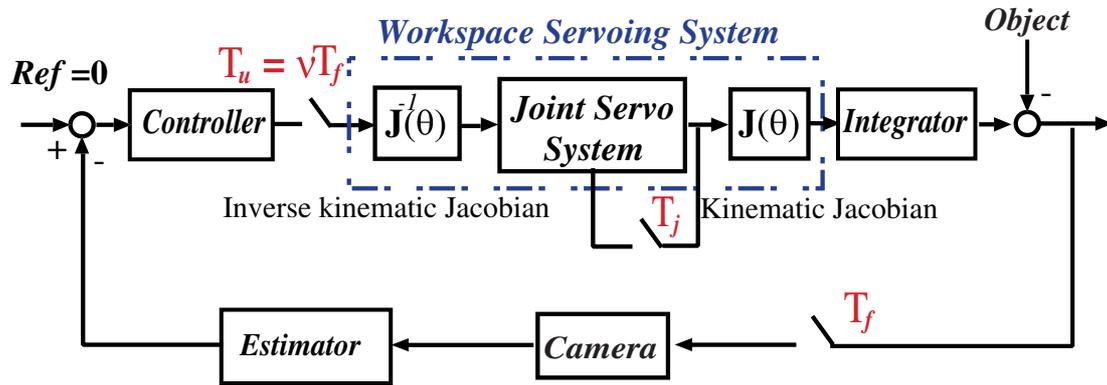


Figure 3.3: Multirate sampling visual servo system

It is also called single-rate sampling control. In recent years, multirate sampling control have been developed well. The control input period( $\nu_i T_f$ ,  $\nu_i = \frac{i}{N}$ , where  $N$  means the control input is changed  $N$  times during  $T_f$ ) is shorter than the longest period(shown in Fig. 3.4(b)).

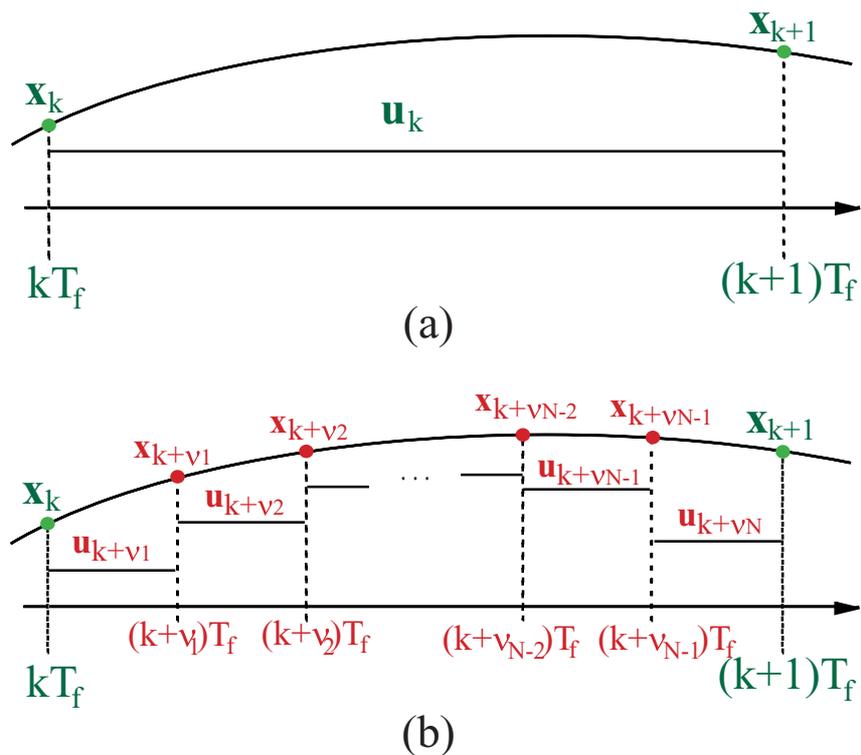


Figure 3.4: (a)Single-rate (b)Multirate sampling control

### 3.4 Diagonalization Problem and Coordinate Transformation Problem

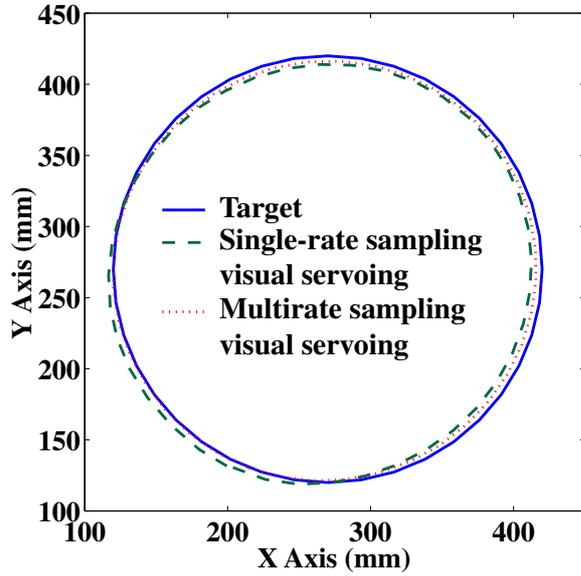


Figure 3.5: Low speed tracking  $\pi/3$ [rad/s]

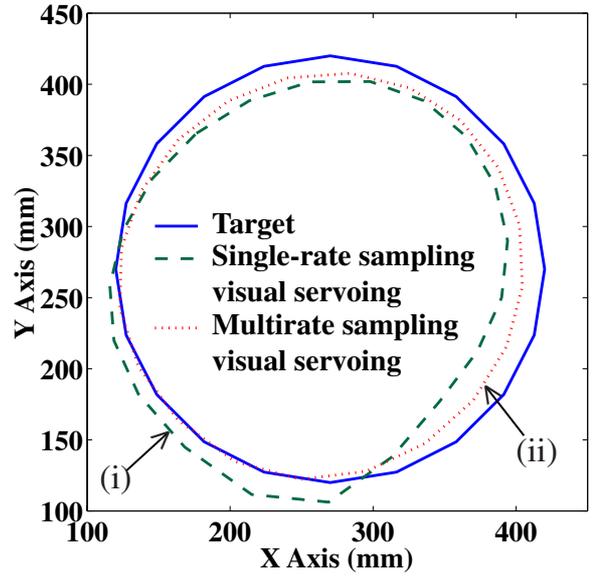


Figure 3.6: High speed tracking  $1\pi$ [rad/s]

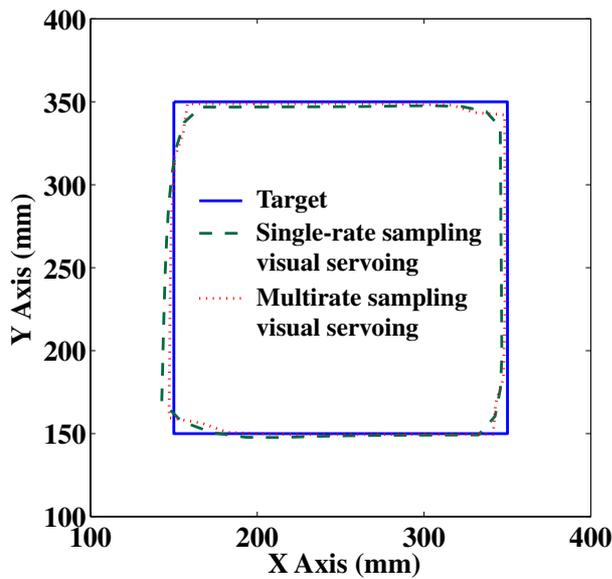


Figure 3.7: Low speed tracking 200[mm/s]

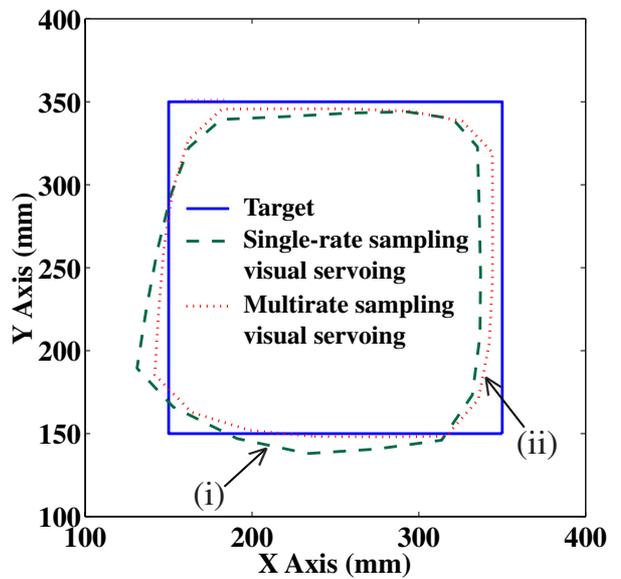


Figure 3.8: High speed tracking 400[mm/s]

In the previous subsection, we know if  $\mathbf{H}$  can be diagonalized if

$$\lim_{z \rightarrow 1} \mathbf{H}(z) = \mathbf{I} \quad (3.9)$$

We can design controller for each joint individually and only need to calculate it once off-line. But if it is not, the controller design problem will become complicated and the calculation load will become drastically because it is updating every sampling period [5] [6]. A usual argument against GPC is its demand for processing power, especially in multi-axial-joint robot control. The higher calculation load it has, the lower control frequency it would have.

### 3.4.1 Conventional Visual Servoings

In the past, in order to avoid this problem, the method used to diagonalize  $\mathbf{H}(z)$  is to consider a sampling interval that is short enough (Use a high sampling but expensive machine vision system or perform low speed tracking) so that the target and the robot moves only a little during this interval and assume that  $\mathbf{J}$  is constant between the two sampling instants. That is to say, if  $\|\theta_k - \theta_{k-1}\| \approx 0$ , then  $\mathbf{H}(z)$  can be diagonalized appropriately (see Fig. 3.5 and Fig. 3.7).

Actually, in industrial applications, i.e. a material handling robot workstation, wide-area operation and high speed moving are always the case. What we are interested in is which problems will occur while performing high speed visual tracking without using high sampling rate machine vision. These problems will be discussed later. Now, let us define two relational expressions in continuous form as follows:

$$\textit{Diagonalization} : \lim_{s \rightarrow \infty} \mathbf{J}(\theta) \mathbf{P}(s) \mathbf{J}^{-1}(\theta) = \mathbf{I} \quad (3.10)$$

$$\textit{Coordinate transformation} : \hat{\theta}^{ref} = \mathbf{J}^{-1}(\theta) \Delta \hat{\mathbf{s}}^{ref} \quad (3.11)$$

If (3.10) can be satisfied, we consider that the workspace servo system can be diagonalized. If (3.11) can be satisfied, we consider that the control input of joint servo system can be transformed correctly from the calculated control input of vision loop.

### Single-rate Sampling Visual Servoing

The block diagram of workspace servo system is shown in Fig. 3.9.  $\mathbf{H}(z)$  cannot be diagonalized to  $\mathbf{I}$  due to  $\|\theta_k - \theta_{k-1}\| \neq 0$ . The control accuracy will become worse while performing high speed tracking (see (i) in Fig. 3.6 and Fig. 3.8).

$$\textit{Diagonalization} : \lim_{z \rightarrow 1} \mathbf{J}(\theta_k) \mathbf{P}(z) \mathbf{J}^{-1}(\theta_{k-1}) \neq \mathbf{I} \quad (3.12)$$

$$\textit{Coordinate transformation} : \hat{\theta}_{k-1}^{ref} = \mathbf{J}^{-1}(\theta_{k-1}) \Delta \hat{\mathbf{s}}_{k-1}^{ref} \quad (3.13)$$

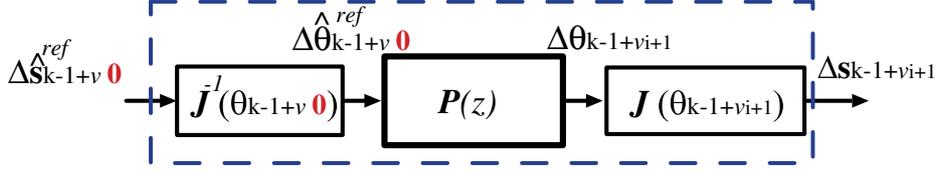


Figure 3.9: Workspace servo system without updating angle

### Multirate Sampling Visual Servoing

To update the  $J^{-1}$  and  $J$  using joint servo loop sampling rate is another approach. The block diagram of workspace servo system is shown in Fig. 3.10.

$$\textit{Diagonalization} : \lim_{z \rightarrow 1} \mathbf{J}(\theta_{k-1+\nu_{i+1}}) \mathbf{P}(z) \mathbf{J}^{-1}(\theta_{k-1+\nu_i}) \approx \mathbf{I} \quad (3.14)$$

Here,  $\nu_i$  means  $i \times \nu T_f$  where  $\nu T_f$  is the control period of joint loop.

Although diagonalization is satisfied in order to solve the serious diagonalized problem (discussed in 4.1.1), oppositely inaccurate coordinate transformation will occur during  $T_f$  (see (ii) in Fig. 3.6 and 3.8). Therefore, the positioning error will accumulate and finally result in poor control accuracy.

$$\textit{Inaccurate coordinate transformation} : \hat{\boldsymbol{\theta}}_{k-1+\nu_i}^{ref} \neq \mathbf{J}^{-1}(\theta_{k-1+\nu_i}) \Delta \hat{\mathbf{s}}_{k-1}^{ref} \quad (3.15)$$

$$\textit{Accurate coordinate transformation} : \hat{\boldsymbol{\theta}}_{k-1}^{ref} = \mathbf{J}^{-1}(\theta_{k-1}) \Delta \hat{\mathbf{s}}_{k-1}^{ref} \quad (3.16)$$

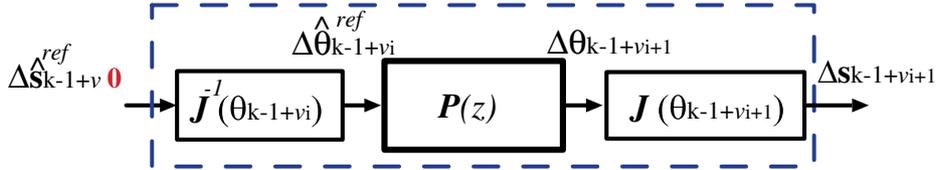


Figure 3.10: Workspace servo system with updating angle

Some researchers considered the delay problem of visual servoing and incorporated prediction method into their systems. The block diagram of workspace servo system is shown in Fig. 3.11. We can know there still exists the inaccurate coordinate transformation problem because of the inaccurate coordinate transformation.

$$\textit{Inaccurate coordinate transformation} : \hat{\boldsymbol{\theta}}_{k+\nu_i}^{ref} \neq \mathbf{J}^{-1}(\theta_{k+\nu_i}) \Delta \hat{\mathbf{s}}_k^{ref} \quad (3.17)$$

$$\textit{Accurate coordinate transformation} : \hat{\boldsymbol{\theta}}_k^{ref} = \mathbf{J}^{-1}(\theta_k) \Delta \hat{\mathbf{s}}_k^{ref} \quad (3.18)$$

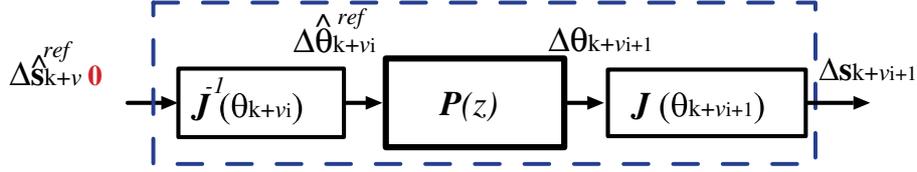


Figure 3.11: Workspace servo system with updating angle and prediction

### 3.4.2 Alternative Visual Servoing

Up to now, we have mentioned about some problems that will occur while performing a high speed visual servo control. If we can design a control scheme as Fig. 3.12, the following two relational expressions will be satisfied, we can solve both the diagonalized problem and coordinate transformation problem appropriately at the same time.

$$\text{Diagonalization} : \lim_{z \rightarrow 1} \mathbf{J}(\theta_{k-1+\nu_{i+1}}) \mathbf{P}(z) \mathbf{J}^{-1}(\theta_{k-1+\nu_i}) \approx \mathbf{I} \quad (3.19)$$

$$\text{Coordinate transformation} : \hat{\theta}_{k-1+\nu_i}^{ref} = \mathbf{J}^{-1}(\theta_{k-1+\nu_i}) \Delta \hat{\mathbf{s}}_{k-1+\nu_i}^{ref} \quad (3.20)$$

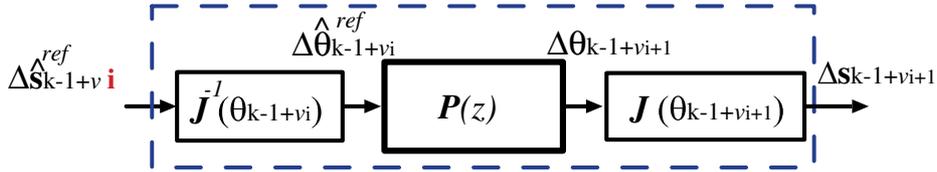


Figure 3.12: Workspace servo system with updating angle and intersample prediction

However it a new problem exists, that is how to obtain the intersample control inputs. If we neglect the delay problem of visual servoing, interpolation[4] can be a good method to solve this problem. However the delay of machine vision is an important problem that cannot be neglected, since tracking task will fail if the target is out of field of the camera. That is to say, solving this problem means we need an intersample predictor to implement this idea.

## 3.5 Summary

The approach used to model the visual servo loop was described. The multirate sampling characteristics of visual servo system was introduced. The delay of camera and the effects of multirate sampling characteristics were taken into consideration. Then computer simulations were carried out to examine that past conventional visual servo control schemes exhibit the diagonalization problem and the coordinate transformation

problem. Finally, the alternative design idea of visual servoing was introduced. The problem, the need of an intersample predictor that occurs in doing the alternative design was also mentioned.



# Chapter 4

## Intersample Prediction Control Schemes Based on Multirate GPC for High-Speed Tracking Tasks

### 4.1 Abstract

In this chapter, the conventional visual servo control scheme and the conventional visual servo control scheme with prediction are defined first for comparison. The concepts of proposed approaches, fastrate controller with intersample predictor and multirate controller with estimator, are described briefly in Section 4.2. Kalman filter[39] is used to estimate target state due to its excellent performance in filtering the noise. The basic Kalman filter and its derivatives are introduced in Section 4.3. By using the lifting technique, the intersample plant model can be established. From this model, fastrate controller and multirate controller can be derived. In Section 4.4, the derivations of the controllers are explained in detail.

### 4.2 The Concepts of Proposed Approaches

In the previous section, we have mentioned about some problems that will occur while performing a high speed visual servo control and an intersample predictor is necessary to implement a perfect visual servoing. In this section we will explain our proposed intersample predictor visual servo control schemes.

Corke[35][36][41] and Hashimoto[21][31] has already studied the delay of vision in view of control theories but the controller they used is PID controller. In the visual servoing[5][6], Gangloff showed drastic improvement of the loop performance with respect

to more classical control strategies like PID type control and the experimental results on a 6 DOF industrial robot are presented that validate the proposed model. However, it needs to update the controller every sampling period. Thus to do a real-time control is difficult because of huge calculation load. In this paper, we design an intersample predictor based on multirate GPC using discrete-time lifting to let the control period of vision loop equal to intersample prediction period. Consequently, our proposed control system(see Fig. 4.3 and Fig. 4.4) based on the idea of perfect visual servoing can satisfy both equation(3.19) and equation(3.20) and then solve the mentioned problems while performing a high-speed visual servo control. Moreover, the calculation load will be reduced drastically because we only have to calculate it once off-line.

Now, let's explain the comparison approach in this paper. The comparison approach in this paper is shown in Fig.4.1. It was proposed by Gangloff et al.[5][6]. In view of the long time delay problem, we introduced a predictor into Fig. 4.1 and modified it as Fig. 4.2.

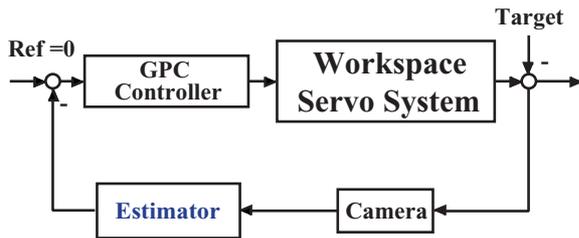


Figure 4.1: Conventional visual servo control scheme

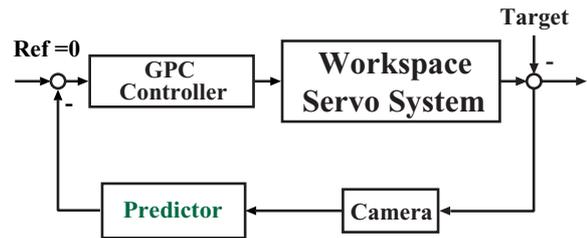


Figure 4.2: Conventional visual servo control scheme with prediction

### 4.2.1 Fastrate controller with Intersample Predictor

The control scheme using fastrate GPC controller with an intersample predictor is proposed. In this work, we used discrete-time lifting to design a fastrate GPC controller and combined it with an intersample predictor to let the control period of vision loop equal to intersample prediction period. The block diagram is shown in Fig. 4.3.

### 4.2.2 Multirate controller with Estimator

In this work, we used discrete-time lifting to design a multirate GPC controller. The use of the predictor in the algorithmic basis of GPC makes this multirate GPC controller

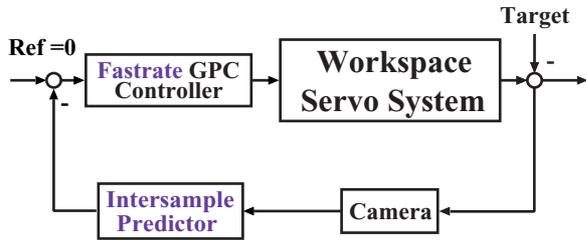


Figure 4.3: Proposed visual servo control scheme-fastrate GPC controller with intersample predictor

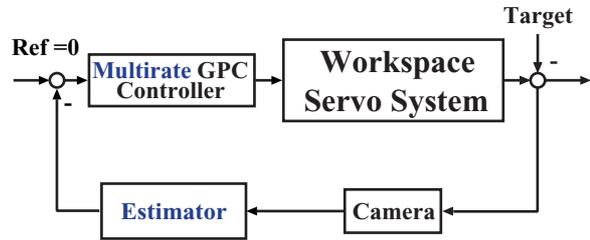


Figure 4.4: Proposed visual servo control scheme-taking multirate GPC controller as intersample predictor

act as an intersample predictor. In this approach, a multirate GPC controller is combined with an estimator to let the control period of vision loop equal to intersample prediction period. The block diagram is shown in Fig. 4.4.

### 4.3 Target State Estimation

The machine vision could occasionally fail to generate any useful data but noise due to variations of the environment illumination, and accidental obstruction of the camera. The uncertainty of visual data comes basically from the high sensitivity of computer vision to its environment and computational errors in image digitization and processing. Image processing algorithms are always time-consuming and there also exists some execution time for the robot end-effector to arrive at a set point. Thus it is inappropriate to use received visual data directly.

Studies on target state estimates based on Luenberger observer,  $\alpha - \beta - \gamma$  filter, Kalman filter, and AR model were carried out by many authors[7]. Kalman filter is used as estimator. Predictive Kalman filter is used as predictor, combined with polynomial interpolation algorithm to establish an intersample predictor. The relations between them are shown in Fig. 4.5. In this paper, we established a 3-ordered dynamical model(see (4.3)) based on Taylor series for the target motion. Then incorporated it into Kalman filter to estimate the target state.

In discrete-time state-space form, the target dynamics consisted of two matrix equations: a transition equation and a measurement equation

#### Signal process:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \omega_k \quad (4.1)$$

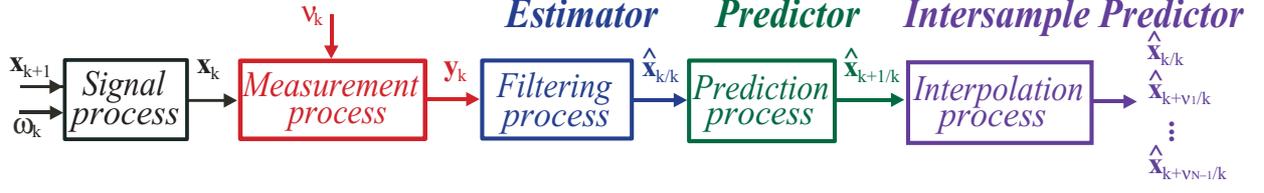


Figure 4.5: Block diagram of Kalman Filter

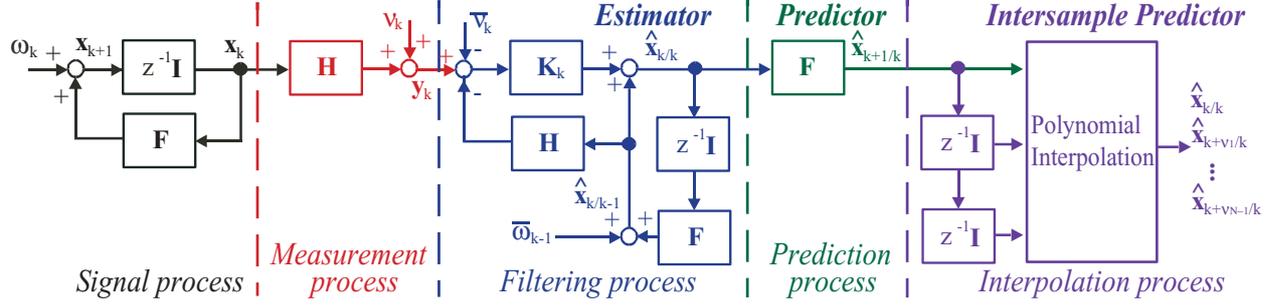


Figure 4.6: Basic structure of Kalman Filter

### Measurement process:

$$y_k = \mathbf{H}\mathbf{x}_k + \nu_k \quad (4.2)$$

The vectors  $\omega_k$  and  $\nu_k$  are the process noise sequence with covariance  $\mathbf{Q}$  and the measurement noise sequence with covariance  $\mathbf{R}$  respectively.

The target motion is unmodeled. Thus the state-transition matrix  $\mathbf{F}$  cannot be known in advance. In this research, we model this matrix according to Taylor series so that the elements of states can be considered as the position, velocity and acceleration of target motion. Furthermore, the target position is chosen as the output for position-based visual servoing. Thus the state-transition matrix  $\mathbf{F}$  and the observation matrix  $\mathbf{H}$  are

$$\mathbf{F} = \begin{bmatrix} 1 & T_f & \frac{T_f^2}{2!} \\ 0 & 1 & T_f \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{H} = [1 \ 0 \ 0] \quad (4.3)$$

where  $T_f$  is the sampling period of vision sensor, the frame period.

For each Cartesian coordinate, one iteration of the Kalman filter algorithm at time step  $k$  is given by

### Filtering process:

State update

$$\hat{\mathbf{x}}_{k/k} = \hat{\mathbf{x}}_{k/k-1} + \mathbf{K}_k[y_k - \mathbf{H}\hat{\mathbf{x}}_{k/k-1}] \quad (4.4)$$

Gain matrix

$$\mathbf{K}_k = \mathbf{P}_{k/k-1} \mathbf{H}^T [\mathbf{H} \mathbf{P}_{k/k-1} \mathbf{H}^T + \mathbf{R}]^{-1} \quad (4.5)$$

Covariance update

$$\mathbf{P}_{k/k} = \mathbf{P}_{k/k-1} - \mathbf{K}_k \mathbf{H} \mathbf{P}_{k/k-1} \quad (4.6)$$

### Prediction process:

State prediction

$$\hat{\mathbf{x}}_{k+1/k} = \mathbf{F} \hat{\mathbf{x}}_{k/k} \quad (4.7)$$

Covariance prediction

$$\mathbf{P}_{k+1/k} = \mathbf{F} \mathbf{P}_{k/k} \mathbf{F}^T + \mathbf{Q} \quad (4.8)$$

where  $\mathbf{K}$  is the Kalman gain matrix,  $\mathbf{P}$  is the error covariance matrix. If we only used the output of filtering process, the optimal target state estimation at time  $k$  is  $\hat{\mathbf{x}}_{k/k}$ . The Kalman filter can be considered as an estimator and used in Fig. 4.1 and Fig. 4.4. If we used the output of prediction process, the optimal target state predicted value at time  $k$  is  $\hat{\mathbf{x}}_{k+1/k}$ . The Kalman filter can be considered as a predictor and be used in Fig. 4.2. It is also called predictive Kalman filter. In this paper, the interpolating polynomial of degree 2 through three past predicted target states,  $\hat{\mathbf{x}}_{k+1/k}$ ,  $\hat{\mathbf{x}}_{k/k-1}$ ,  $\hat{\mathbf{x}}_{k-1/k-2}$  is used to construct an intersample predictor based on Kalman filter. Then, using it we can interpolate the intersample target states for Fig. 4.3. The basic structure of Kalman filter is shown in Fig. 4.6.

## 4.4 Lifting GPC Controller

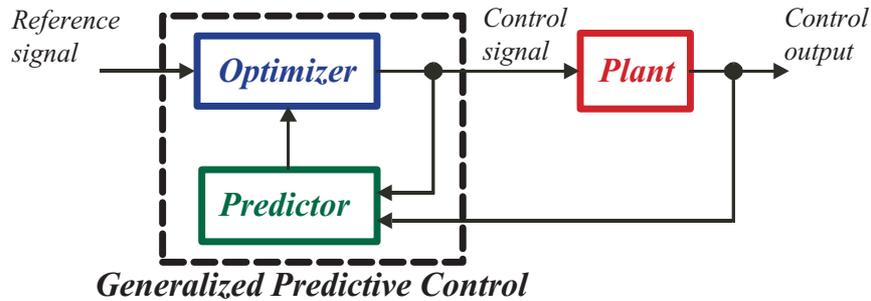


Figure 4.7: Block diagram of GPC

Traditionally, GPC has been derived using a transfer function model through Diophantine equations [17][5][6]. A proof[18] was provided to show that the state space approach is equivalent to polynomial approach. Thus, we consider a state space formulation, which has the advantage that taking discrete-time lifting[20] to be multirate form is straightforward.

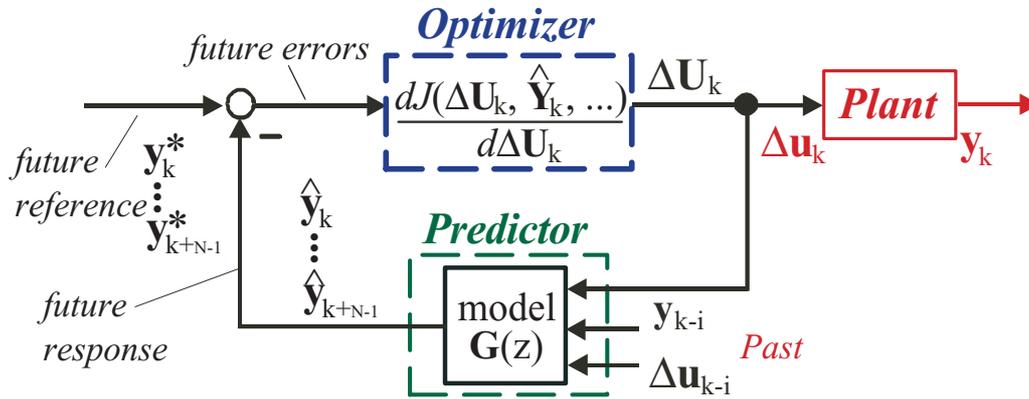


Figure 4.8: Basic structure of single-rate GPC

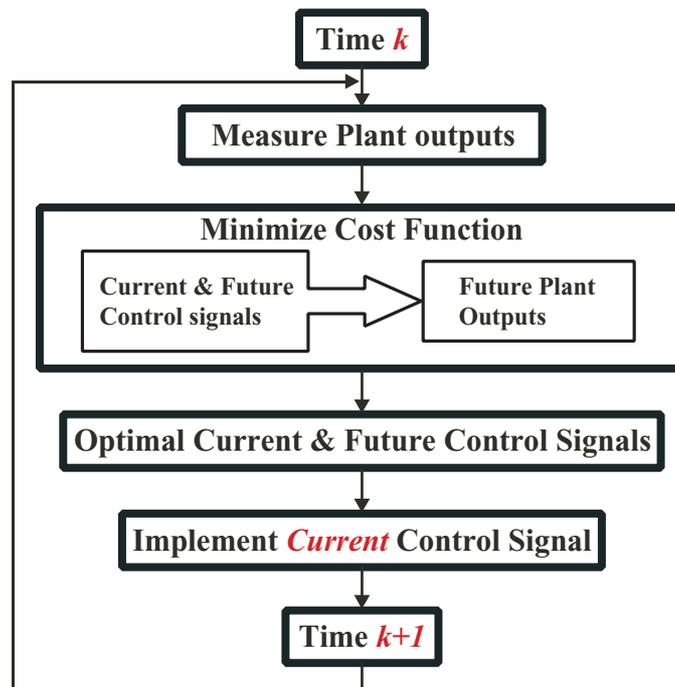


Figure 4.9: Basic algorithm of single-rate GPC

GPC is a simpler version of the LQG controller that uses model predictive method to construct a predictor and uses least squares method to construct an optimizer. Thus,

GPC is a control strategy that uses knowledge of the future behavior of the output for calculating the feedback signal(see Fig. 4.7). That is to say, it generates a sequence of future control signals within each sampling interval to optimize the control effort of the controlled system.

Let us lift variables by some sampling time that is shorter than frame period, then the lifted GPC will be able to generate a sequence of future intersample control signals within each sampling interval. Now we will introduce the lifted GPC algorithm that is established by plant model, prediction formulation and control law. In order to compare with the conventional single-rate GPC in state space approach, its basic structure and algorithm are shown in Fig. 4.8 and Fig. 4.9.

### Intersample plant model:

Consider a controllable and observable LTI system represented in discrete time by

$$\begin{aligned}\mathbf{x}_{k+\nu_{j+1}} &= \mathbf{A}\mathbf{x}_{k+\nu_j} + \mathbf{B}\Delta\mathbf{u}_{k+\nu_j} + \mathbf{E}\mathbf{d}_{k+\nu_j} \\ \mathbf{y}_{k+\nu_j} &= \mathbf{C}\mathbf{x}_{k+\nu_j}\end{aligned}\quad (4.9)$$

where  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  are derived from  $\mathbf{G}(z)$  in equation(3.5) using sampling time  $\nu T_f$ ,  $\nu = T_f/N$ ,  $\nu_i = \frac{T_f}{N} \times i$ ,  $N$  is the intersample prediction input times during  $T_f$ . Here we consider the target motion act as the disturbance,  $\mathbf{d}_{k+\nu_j}$ , included in system.

### Prediction formulation:

Let's solve the state space equations recursively, the following  $j$ -step intersample future output prediction can be derived.

$$\mathbf{y}_{k+\nu_j} = \mathbf{C}\mathbf{A}^j\mathbf{x}_k + \sum_{i=0}^{j-1} \mathbf{C}\mathbf{A}^{j-i-1}\mathbf{B}\Delta\mathbf{u}_{k+\nu_i} + \sum_{i=0}^{j-1} \mathbf{C}\mathbf{A}^{j-i-1}\mathbf{E}\mathbf{d}_{k+\nu_i}\quad (4.10)$$

Since the spectral properties of disturbances are unknown(arbitrary target motion), we remove them and consider them as the variance of system. Thus, the minimum variance estimator of  $\mathbf{y}_{k+\nu_j}$  is given by

$$\hat{\mathbf{y}}_{k+\nu_j} = \mathbf{C}\mathbf{A}^j\mathbf{x}_k + \sum_{i=0}^{j-1} \mathbf{C}\mathbf{A}^{j-i-1}\mathbf{B}\Delta\mathbf{u}_{k+\nu_i}\quad (4.11)$$

### Control law:

The objective of the GPC is to synthesize a set of optimal control input increments  $\Delta\mathbf{u}_{k+\nu_{i-1}}$ ,  $i = 1, \dots, N_u$ . Using equation(4.11), we can obtain the following cost function:

$$\begin{aligned}J &= \sum_{j=N_1}^{N_2} \mathbf{e}_{k+\nu_j}^T \mathbf{e}_{k+\nu_j} + \lambda \sum_{i=1}^{N_u} \Delta\mathbf{u}_{k+\nu_{i-1}}^T \Delta\mathbf{u}_{k+\nu_{i-1}} \\ &= (\hat{\mathbf{Y}}_k - \mathbf{Y}_k^*)^T (\hat{\mathbf{Y}}_k - \mathbf{Y}_k^*) + \lambda \Delta\mathbf{U}_k^T \Delta\mathbf{U}_k\end{aligned}\quad (4.12)$$

where  $\mathbf{e}_{k+\nu_j}$  is the future  $j$ -step intersample predicted tracking error  $\mathbf{e}_{k+\nu_j} = \hat{\mathbf{y}}_{k+\nu_j} - \mathbf{y}_{k+\nu_j}^*$ ,  $N_1$ ,  $N_2$  and  $N_u$  are the minimum, maximum and control costing horizons respectively,  $\lambda$  is the constant control input costing factor and  $\mathbf{y}_{k+\nu_j}^*$  is the future reference.

• **Fastrate GPC Controller**

By minimizing cost function, we can obtain the following optimal control increments:

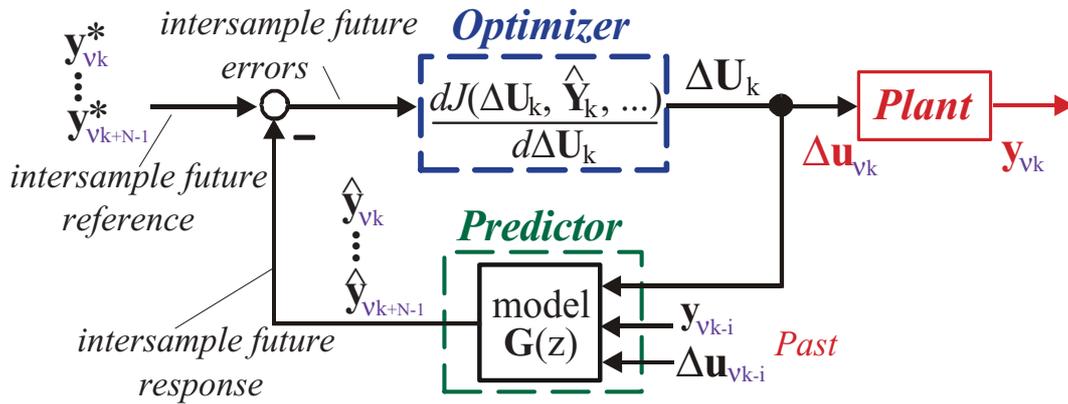


Figure 4.10: Basic structure of fastrate GPC

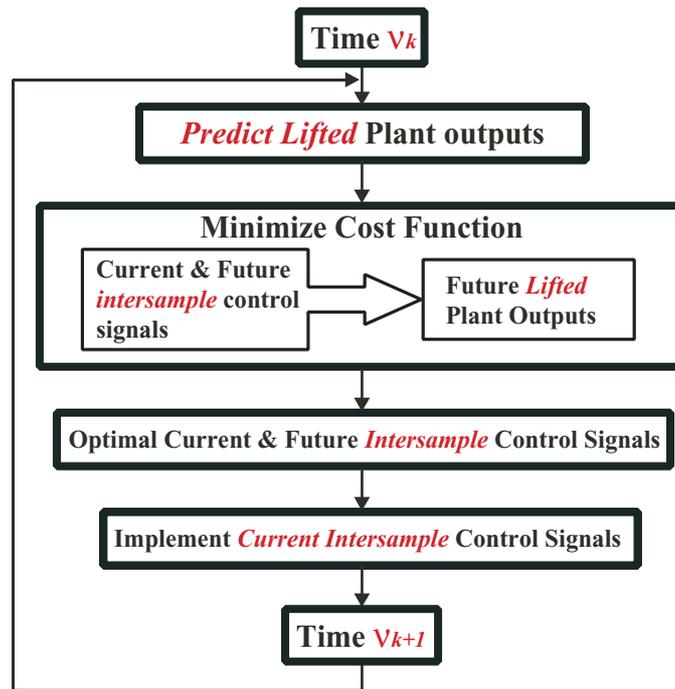


Figure 4.11: Basic algorithm of fastrate GPC

$$\Delta \mathbf{U}_k = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{Y}_k^* - \Phi \hat{\mathbf{x}}_{\nu k}) \quad (4.13)$$

where

$$\hat{\mathbf{Y}}_k = [ \hat{\mathbf{y}}_{\nu_{k+N_1}}, \dots, \hat{\mathbf{y}}_{\nu_{k+N_2}} ]^T \quad (4.14)$$

$$\Delta \mathbf{U}_k = [ \Delta u_{\nu_k}, \dots, \Delta u_{\nu_{k+N_u-1}} ]^T \quad (4.15)$$

$$\mathbf{Y}_k^* = [ \mathbf{y}_{\nu_{k+N_1}}^*, \dots, \mathbf{y}_{\nu_{k+N_2}}^* ]^T \quad (4.16)$$

Since the camera configuration in our work is end-effector mounted, the reference should always be 0 for target tracking,  $\mathbf{Y}_k^* = \mathbf{0}$ . The same as standard single rate GPC, fastrate GPC also only used the first element,  $\Delta u_{\nu_k}$  of the calculated control input matrix,  $\Delta \mathbf{U}_k$ . In this work,  $\Delta u_{\nu_k}$  are applied to the plant for performing intersample control. Its basic structure and algorithm are shown in Fig. 4.10 and Fig. 4.11.

$$\Delta u_{\nu_k} = [ 1, 0, \dots, 0 ] \Delta \mathbf{U}_k \quad (4.17)$$

#### • Multirate GPC Controller

By minimizing cost function, we can obtain the following optimal control increments:

$$\Delta \mathbf{U}_k = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{Y}_k^* - \Phi \mathbf{x}_k) \quad (4.18)$$

where

$$\hat{\mathbf{Y}}_k = [ \hat{\mathbf{y}}_{k+\nu_{N_1}}, \dots, \hat{\mathbf{y}}_{k+\nu_{N_2}} ]^T \quad (4.19)$$

$$\Delta \mathbf{U}_k = [ \Delta u_k, \dots, \Delta u_{k+\nu_{N_u-1}} ]^T \quad (4.20)$$

$$\mathbf{Y}_k^* = [ \mathbf{y}_{k+\nu_{N_1}}^*, \dots, \mathbf{y}_{k+\nu_{N_2}}^* ]^T \quad (4.21)$$

Since the camera configuration in our work is end-effector mounted, the reference should always be 0 for target tracking,  $\mathbf{Y}_k^* = \mathbf{0}$ . Then the intersample optimal control inputs can be obtained by the following equations.

$$\Delta u_k = [ 1, 0, \dots, 0 ] \Delta \mathbf{U}_k \quad (4.22)$$

$$\Delta u_{k+\nu_1} = [ 0, 1, \dots, 0 ] \Delta \mathbf{U}_k \quad (4.23)$$

⋮

$$\Delta u_{k+\nu_{N-1}} = [ 0, 0, \dots, 1 ] \Delta \mathbf{U}_k \quad (4.24)$$

In standard single-rate GPC controller, only the first element,  $\Delta u_k$  is used to perform control(see Fig. 4.8). However, in this work,  $\Delta u_k \sim \Delta u_{k+\nu_{N-1}}$  are applied sequentially to the plant for performing intersample controls. Its basic structure and algorithm are

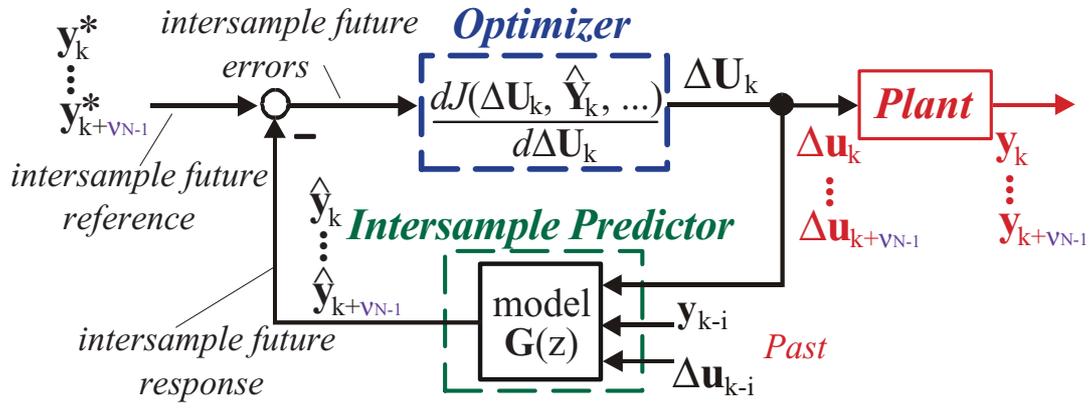


Figure 4.12: Basic structure of multirate GPC

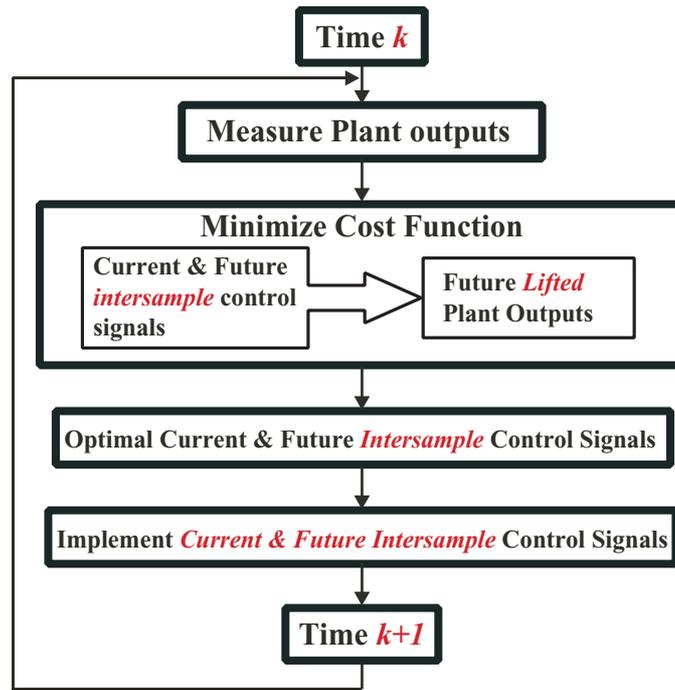


Figure 4.13: Basic algorithm of multirate GPC

shown in Fig. 4.12 and Fig. 4.13.

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{CA}^{N_1-1}\mathbf{B} & \dots & \mathbf{CB} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{CA}^{N_u-2}\mathbf{B} & \ddots & \ddots & \ddots & \ddots & 0 \\ \mathbf{CA}^{N_u-1}\mathbf{B} & \ddots & \ddots & \ddots & \ddots & \mathbf{CB} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{CA}^{N_2-1}\mathbf{B} & \dots & \dots & \dots & \dots & \mathbf{CA}^{N_2-N_u}\mathbf{B} \end{bmatrix} \quad (4.25)$$

(4.26)

$$\Phi = \begin{bmatrix} \mathbf{CA}^{N_1} \\ \mathbf{CA}^{N_1+1} \\ \vdots \\ \mathbf{CA}^{N_2} \end{bmatrix} \quad (4.27)$$

## 4.5 Summary

In the comparison, the conventional visual servo control scheme and the conventional visual servo control scheme with prediction were defined. The concepts of proposed approaches, fastrate controller with intersample predictor and multirate controller with estimator, were described briefly in Section 4.2. The basic Kalman filter and its derivatives for target state estimation were introduced in Section 4.3. Their relationships were also introduced. The algorithms based on lifting technique used to derive fastrate controller and multirate controller were explained. Three pairs of basic structure and basic algorithm figures showed the relations and differences between single-rate GPC controller, fastrate GPC controller and multirate GPC controller. The detailed mathematical model of multirate GPC derived by discrete-time lifting was also explained.



# Chapter 5

## Verifications of the Effects on Proposed Schemes

### 5.1 Abstract

In this chapter, computer simulations and experiments using a 2-link DD robot are carried out to examine the effectiveness of the proposed methods. While performing experiments, the effects of noise in prediction control is quite remarkable. So two pairs of simulations (without noise and with noise) are carried out in order to discuss this problem. This research focuses on unknown target motion, thus two kinds of target motions (circle and square) are performed both in simulations and experiments to show its expand in application.

### 5.2 Simulation Verifications

In the following simulations, the parameter values selected for the GPC controller are  $\lambda = 1$ ,  $N_1 = 1$ ,  $N_2 = 10$ ,  $N_u = 5$  and  $N = 5$ . In order to emphasize the effectiveness of prediction control, two kinds of visual tracking tasks, circular target motion tracking and square target motion tracking are performed. Generally speaking, we can have good dynamic prediction performance by weighting covariance but the stochastic performance will deteriorate due to the amplification of measurement noise. We need to discuss the effects of noise in prediction control. So the simulations are also performed with noise and without noise.

### 5.2.1 Circular Target Motion Tracking without noise

In this simulation, the camera mounted on the 2-link DD robot manipulator tracks a moving target. The initial position of the target is (420, 270), target will start a circular movement with  $radius = 150[mm]$ ,  $\omega = 1\pi[rad/s]$  at  $t = 2.0$ . The sampling period of vision sensor is  $100[ms]$ .  $\iota(\theta)$  is given by

$$\iota(\theta) = \frac{f}{z} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \quad (5.1)$$

where  $\theta =: \theta_1 + \theta_2$ .

The conventional approach is the visual control scheme proposed in [5] and [6], incorporated with a Kalman filter. Moreover, the conventional approach with prediction is also the visual control scheme proposed in [5] and [6], but incorporated with a predictive Kalman filter. In order to simplify the discussion of the effects of different control schemes in coordinate transformation problem, the parameters we adjusted to let these approaches have the same prediction performance(see Fig. 5.2 and Fig. 5.3). Fig. 5.1 shows that although the conventional approach with prediction has higher prediction performance than conventional approach there also exists the coordinate transformation problem. Furthermore, the proposed fastrate and multirate approach have much higher control accuracy than the conventional approach with prediction.

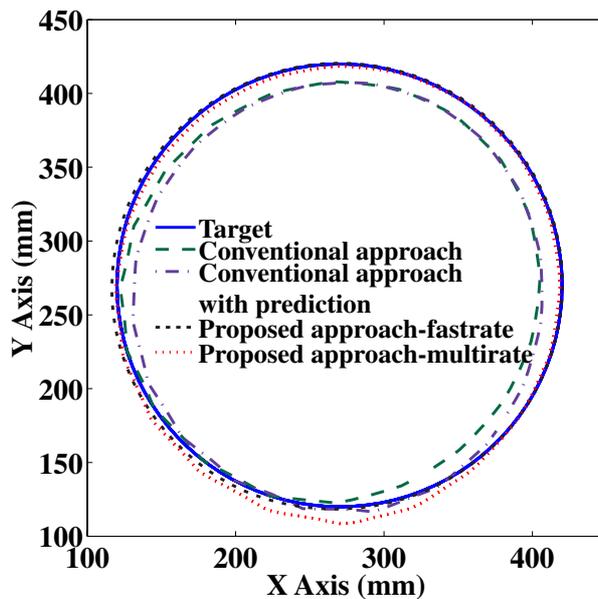


Figure 5.1: Tracking trajectory for circular target motion(simulation without noise)

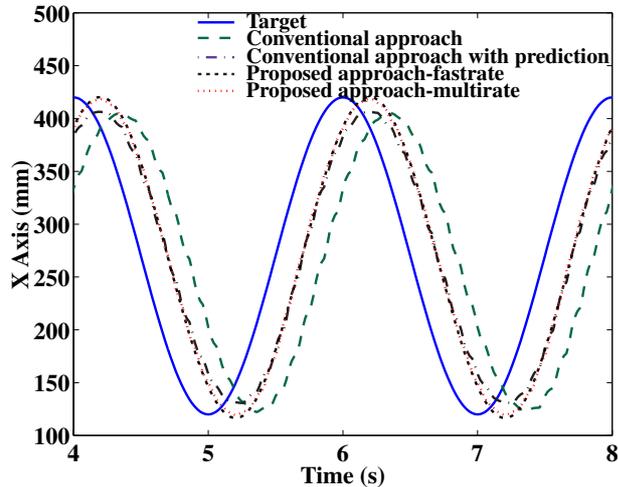


Figure 5.2: Tracking response for circular target motion(simulation without noise)

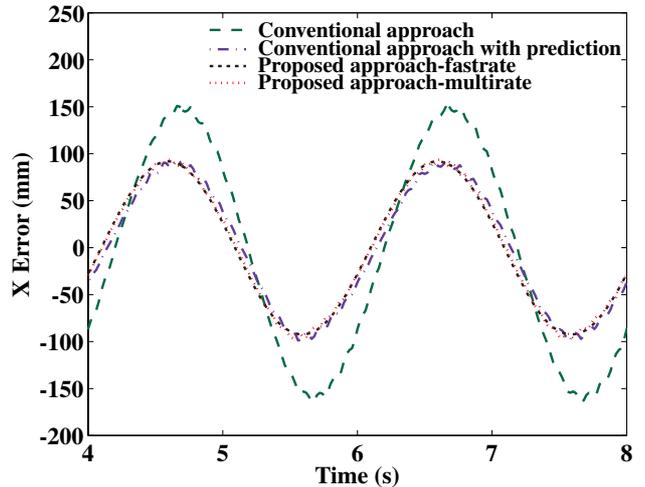


Figure 5.3: Tracking error for circular target motion(simulation without noise)

### 5.2.2 Square Target Motion Tracking without noise

In this simulation, the initial position of the target is  $(350, 150)$ , target will start a square movement with  $length = 200[mm]$ ,  $velocity = 200[mm/s]$  at  $t = 2.0$ . The sampling period of vision sensor is  $100[ms]$ .

We adjust the parameters to let these approaches have the same predicted performance(see Fig. 5.5 and Fig. 5.6) in order to emphasize on the coordinate transformation problem. Fig. 5.4 shows that although the conventional approach with prediction have higher prediction performance than conventional approach but there also exists the coordinate transformation problem. Furthermore, the proposed fastrate and multirate approach have much higher control accuracy than the conventional approach with prediction.

### 5.2.3 Circular Target Motion Tracking with noise

In the following two simulations, all the conditions are the same as 5.2.1 but a white noise introduced into the extracted feature. Here we want to discuss the effects of noise in prediction control. First, a comparison simulation is taken to choose the better one from our proposed approaches to be used to compare with the conventional approaches and find the hidden problem in our proposed approaches. Although fastrate approach and multirate approach have almost the same performance while system contain very small noise, Fig. 5.7 and Fig. 5.8 show that taking multirate GPC as an intersample predictor has much higher noise suppression performance than fastrate GPC controller with intersample predictor(using interpolation).

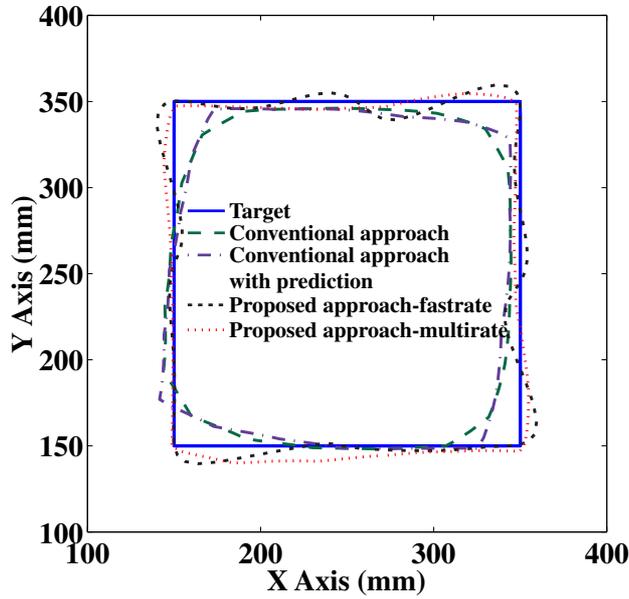


Figure 5.4: Tracking trajectory for square target motion(simulation without noise)

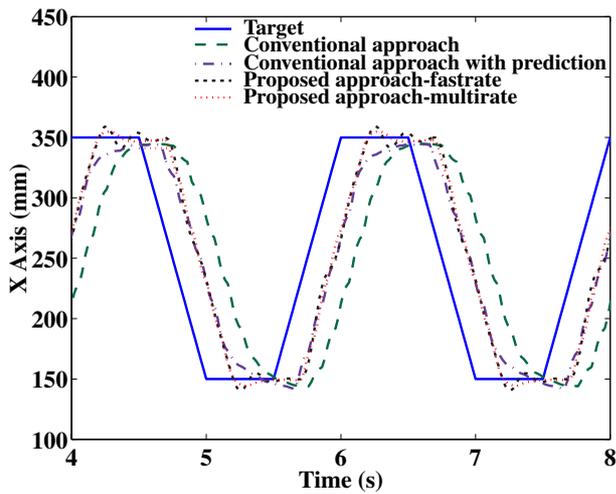


Figure 5.5: Tracking response for square target motion(simulation without noise)

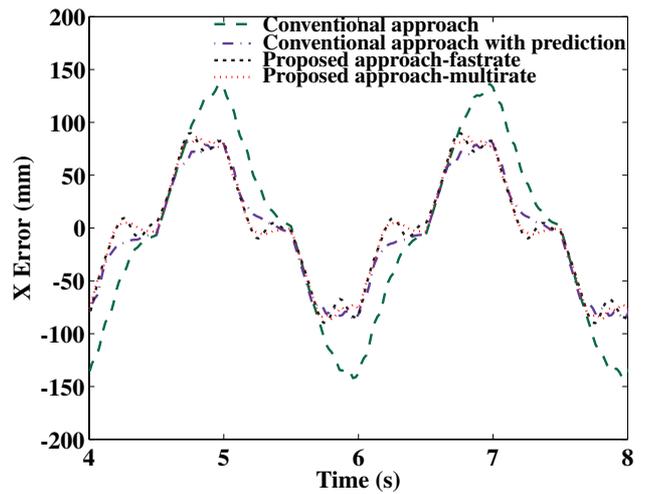


Figure 5.6: Tracking error for square target motion(simulation without noise)

Second, simulations are taken to compare with the conventional approaches. Fig. 5.9, Fig. 5.10 and Fig. 5.11 show that taking multirate GPC as an intersample predictor not only compensates the time delay but also has higher control accuracy than the conventional approach with sacrificing a little noise suppression. On the other hand, proposed approach has higher control accuracy than the conventional approach with prediction without sacrificing the performance of noise suppression.

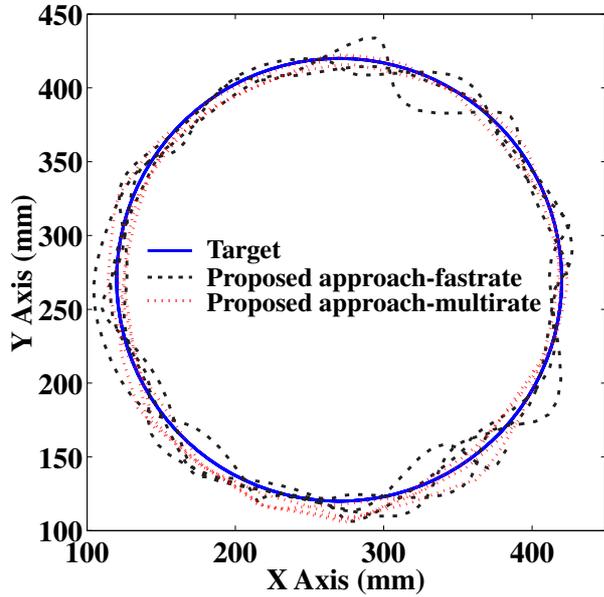


Figure 5.7: Tracking trajectory for circular target motion in proposed approaches(simulation with noise)

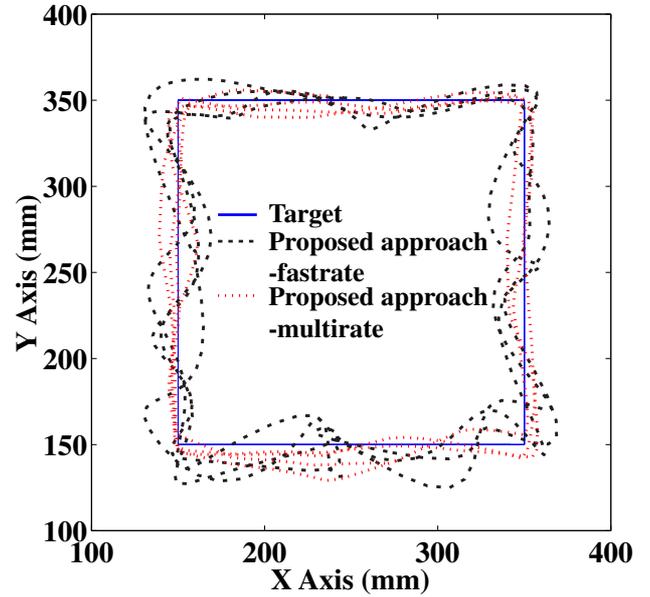


Figure 5.8: Tracking trajectory for square target motion in proposed approaches(simulation with noise)

### 5.2.4 Square Target Motion Tracking with noise

In this simulation, all the conditions are the same as 5.2.2 but a white noise is introduced into the extracted feature. Simulations are taken to compare the effects of noise with the conventional approaches. Fig. 5.12, Fig. 5.13 and Fig. 5.14 show that taking multirate GPC as an intersample predictor not only compensates the time delay but also has higher control accuracy than the conventional approach with sacrificing a little noise suppression. On the other hand, proposed approach has higher control accuracy than the conventional approach with prediction without sacrificing the performance of noise suppression.

## 5.3 Experimental Verifications

In these experiments, the parameter values selected for the GPC controller are  $\lambda = 1$ ,  $N_1 = 1$ ,  $N_2 = 10$ ,  $N_u = 5$  and  $N = 5$ . The same as simulation verifications, two kinds of visual tracking tasks, circular target motion tracking and square target motion tracking are performed in order to emphasize the effectiveness of the prediction control.

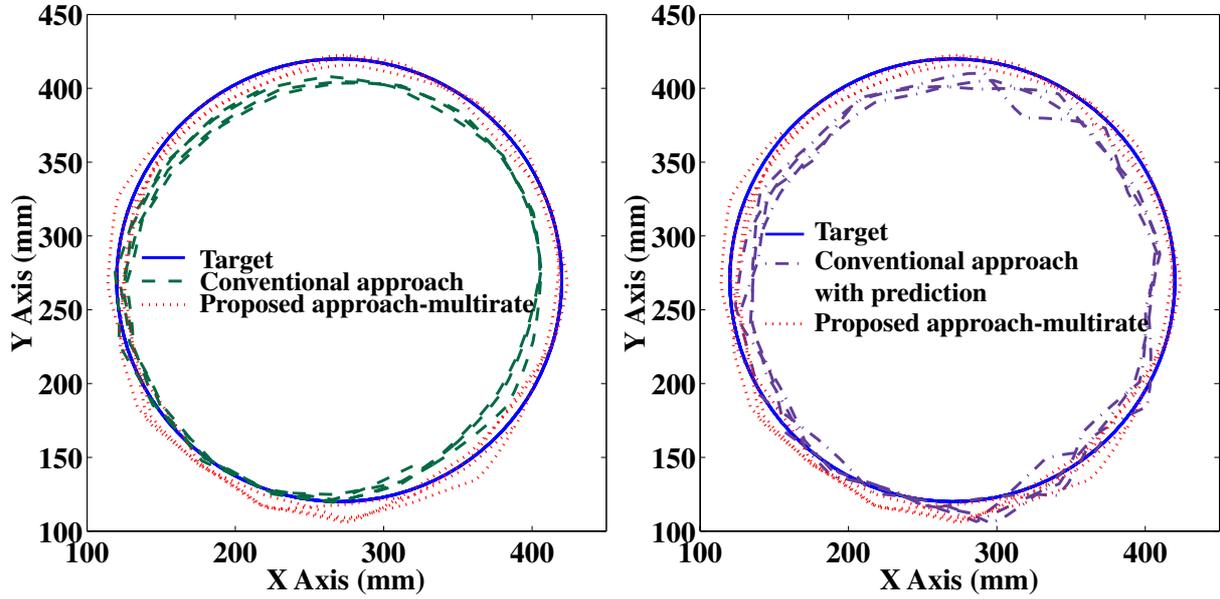


Figure 5.9: Tracking trajectory for circular target motion(simulation with noise)

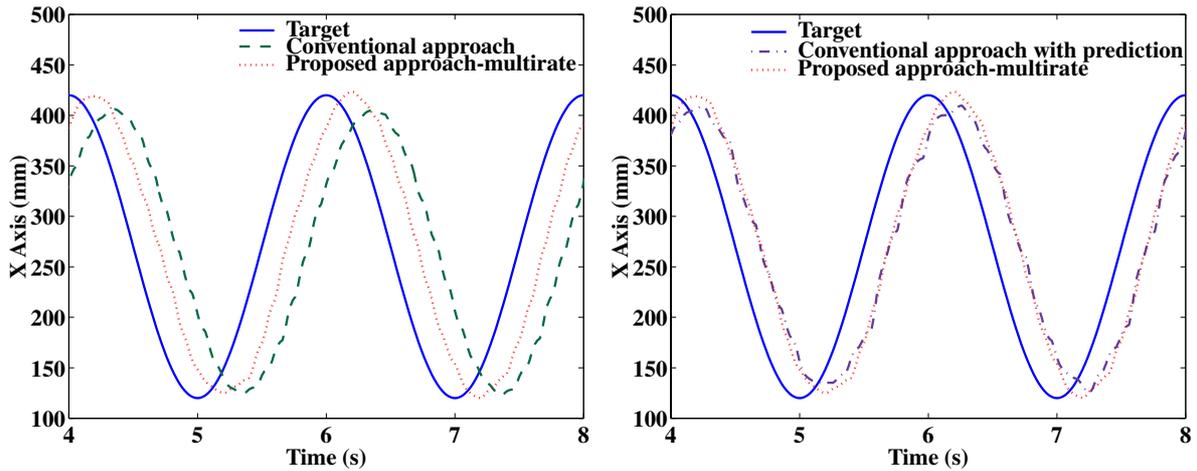


Figure 5.10: Tracking response for circular target motion(simulation with noise)

### 5.3.1 Experimental Hardware and Software

The configuration of experimental setup is shown in Fig. 5.15. The important components of the experimental system are

- Processor and operating system
- Robot control hardware

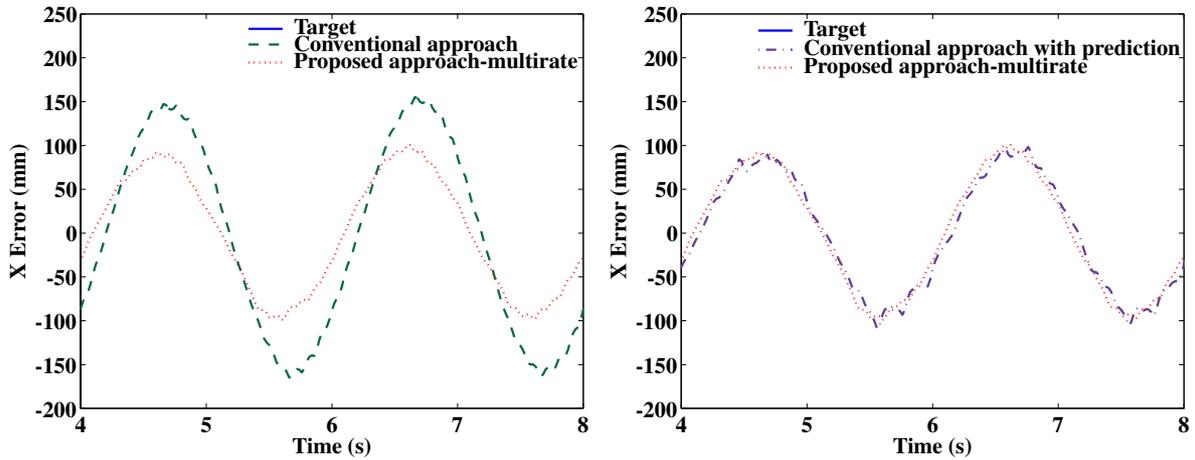


Figure 5.11: Tracking error for circular target motion(simulation with noise)

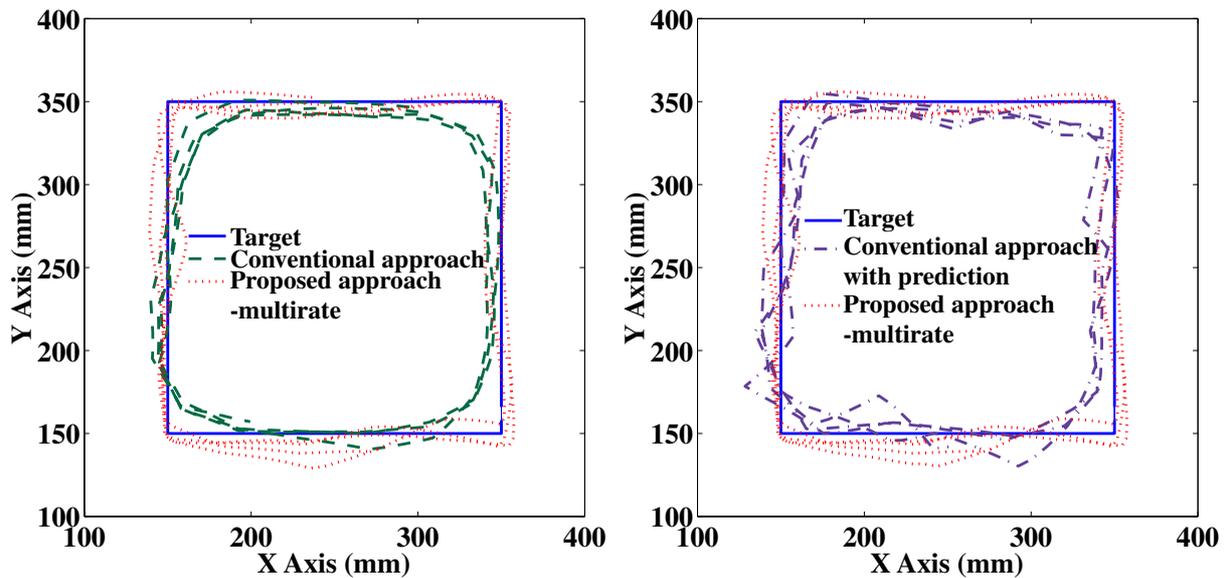


Figure 5.12: Tracking trajectory for square target motion(simulation with noise)

- Machine vision system

The photograph of our experimental equipment is shown in Fig. 5.16. A personal computer(CPU:Celeron 300MHz) is used both for real-time control of servomotors and development of software. In the PC, a D/A converter and a counter board are implemented to output the reference current and to input the motor angle. The servomotors are reluctance motors, where the current is controlled by the motor driver. The encoders generate 38400 pulses per revolution. In order to perform multi-control tasks, RTLinux is installed as the real-time operating system[27]. A schematic of our software is shown in Fig. 5.19. Visual servo loop control task and joint servo loop control task are carried

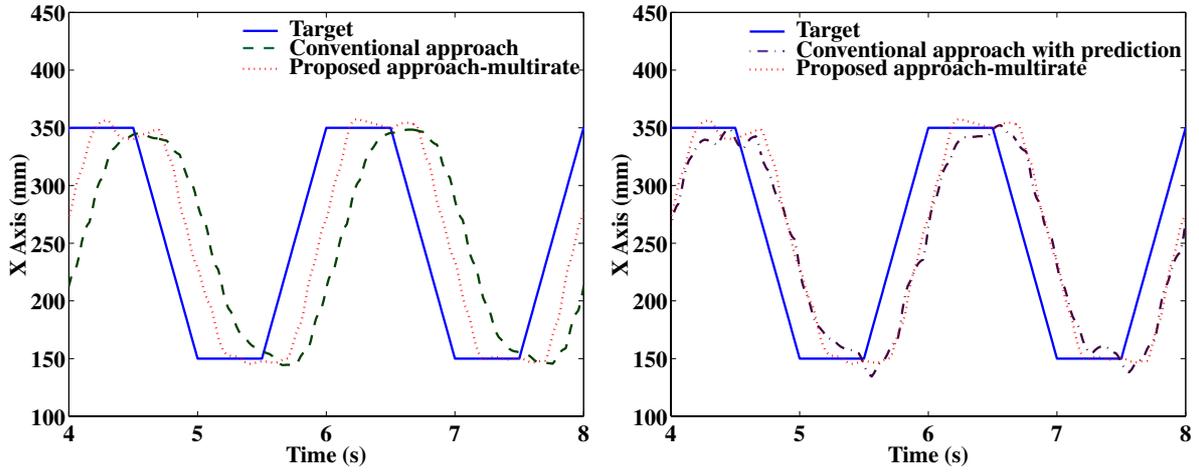


Figure 5.13: Tracking response for square target motion(simulation with noise)

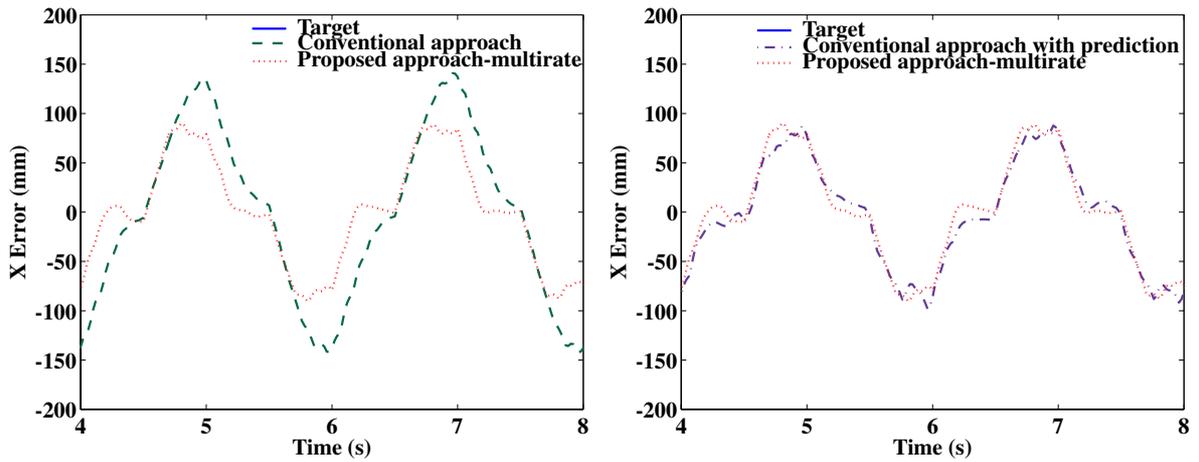


Figure 5.14: Tracking error for square target motion(simulation with noise)

out in RT-kernel space. Image capture and feature extraction are carried out in the user space.

**The image acquisition process:** For our experiments, we used a web camera that only acquired a general frame grabber board with a Analog to Digital Converter(about 9 MPixels/s). Although this device didn't allow us to know precisely which line in the image was currently being acquired, we made an RTlinux program to synchronize the beginning of a sample of the visual servoing controller with the acquisition.

Two frames were used to capture images. When a sample of the visual loop controller began, the last image was already in the frame grabber's memory. So image processing and feature extraction could be performed on the last image while the next image was being acquired. Then, when this acquisition was completed, image processing and feature

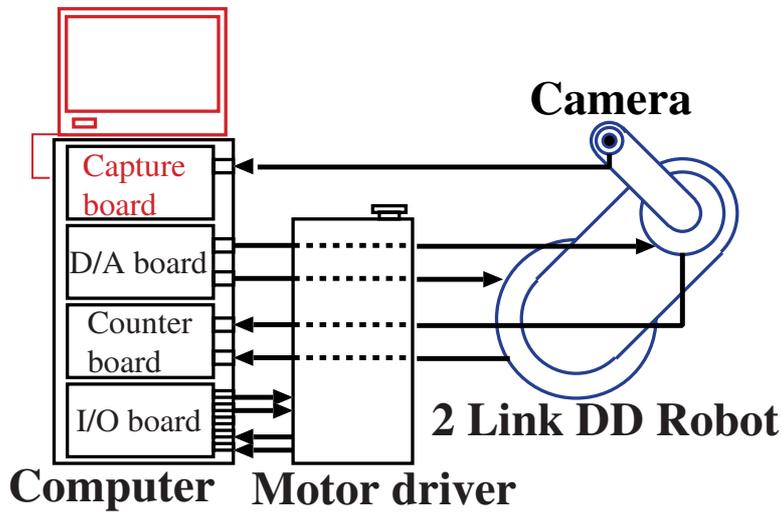


Figure 5.15: Experimental setup

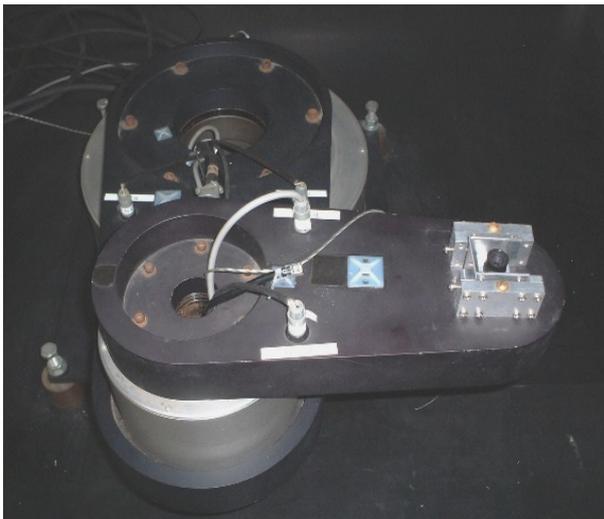


Figure 5.16: Photographs of 2-link DD robot with webcam

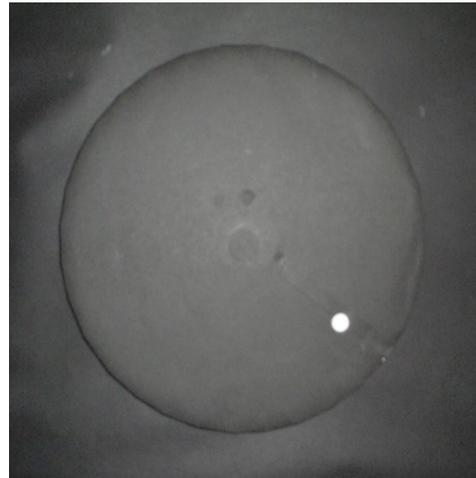


Figure 5.17: Circular motion target

extraction could begin on this image.

We used this strategy in our work to minimize the delay due to image acquisition and image processing. If we considered that the mean sampling instant of an image was at the middle of the integration process (i.e. the middle of an image acquisition if no shutter is used), then the acquisition and image processing of the visual measurement yielded a delay of exactly 2 samples. The first delay was due to the transmission of the image, the second delay was due to the computation of the control inputs. So we slowed down the sampling rate from 30[Hz] to 10[Hz] in order to cover these delays.

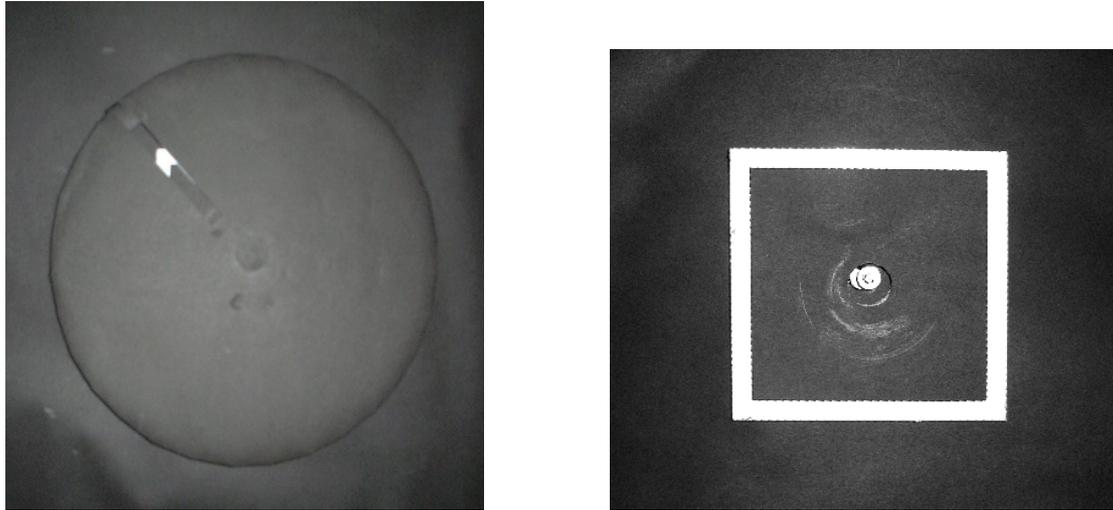


Figure 5.18: Structure of square motion target

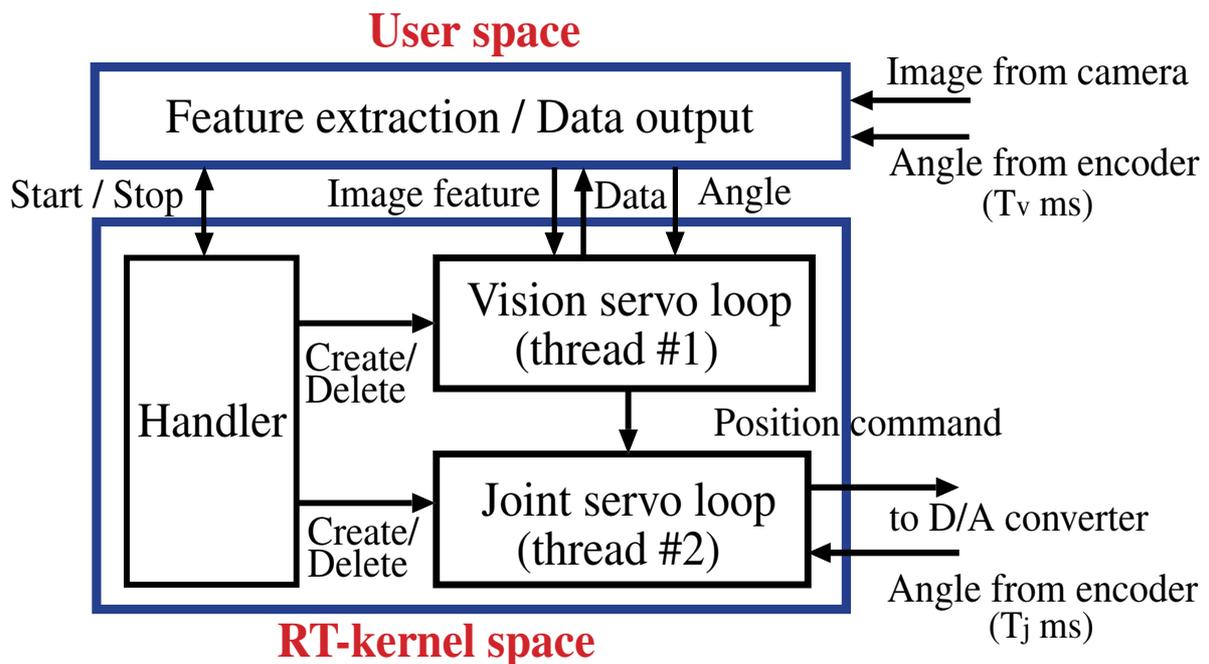


Figure 5.19: Experimental Software

### 5.3.2 Circular Target Motion Tracking

In the experiments, the most common CCD camera: webcam and capture board for PC was used for generating  $200 \times 200$  pixels image per  $100[ms]$ . The target (shown in Fig. 5.17) is rotating at a circular trajectory with constant rotational velocity  $1\pi[rad/s]$  and radius  $100[mm]$ . Fig. 5.20, Fig. 5.21 and Fig. 5.22 show that although the conventional approach with prediction can compensate the delay of vision sensor drastically, the accuracy of

tracking trajectory is sacrificed because it enlarges the measurement noise caused from quantization error. Moreover, Fig. 5.20, Fig. 5.21 and Fig. 5.22 also show the drastic effectiveness of our proposed intersample predictor control scheme. It can compensate the delay of vision sensor without sacrificing the accuracy of tracking trajectory.

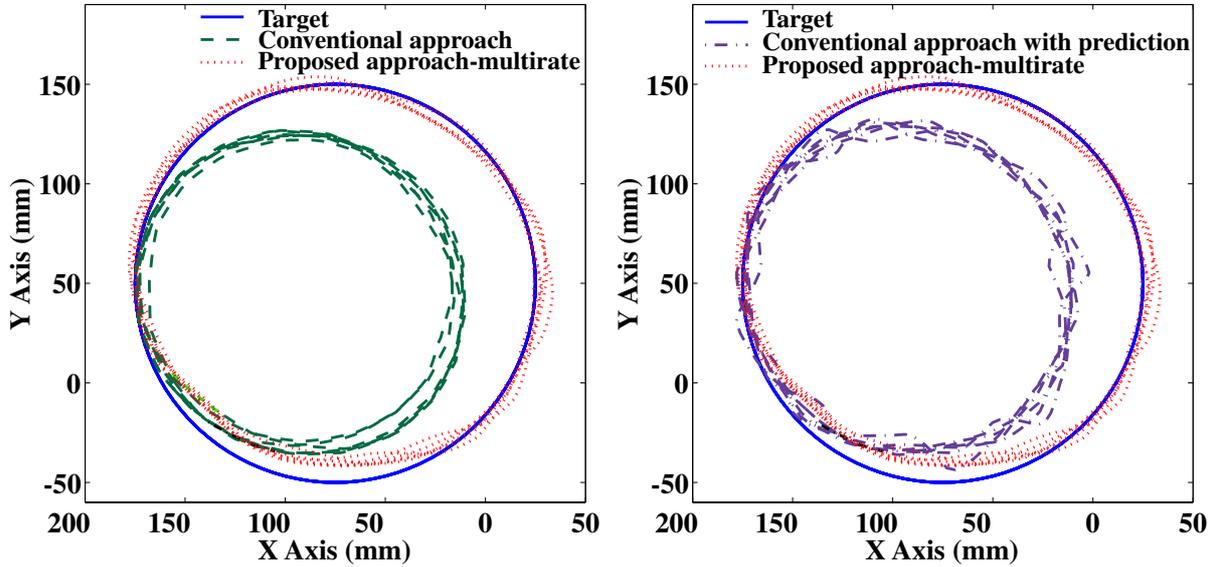


Figure 5.20: Tracking trajectory for circular target motion(experiment)

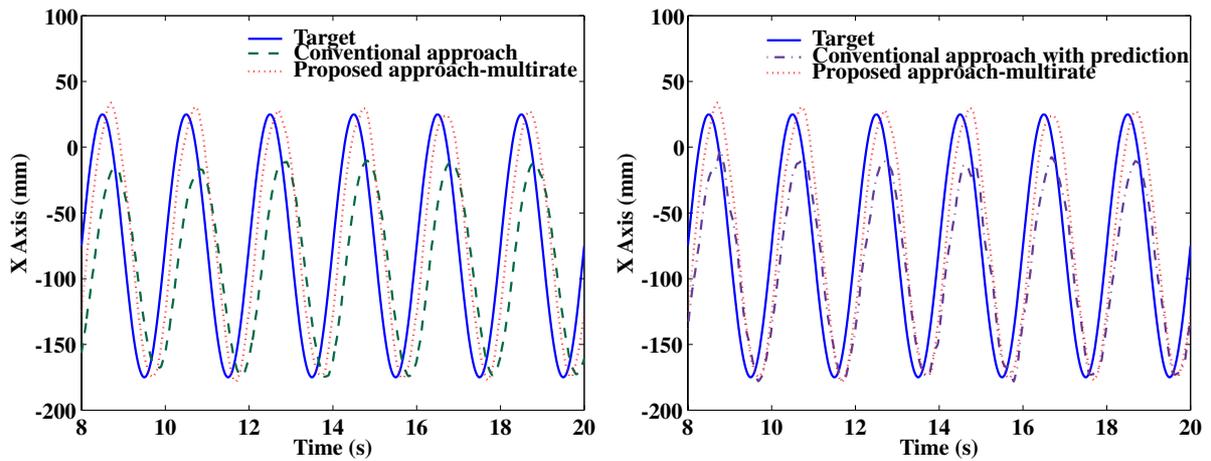


Figure 5.21: Tracking response for circular target motion(experiment)

### 5.3.3 Square Target Motion Tracking

In the experiments, the setup of hardware and software is the same as in 5.3.2. The square motion target (shown in Fig. 5.18) is made by rotating a plate with a rectangular

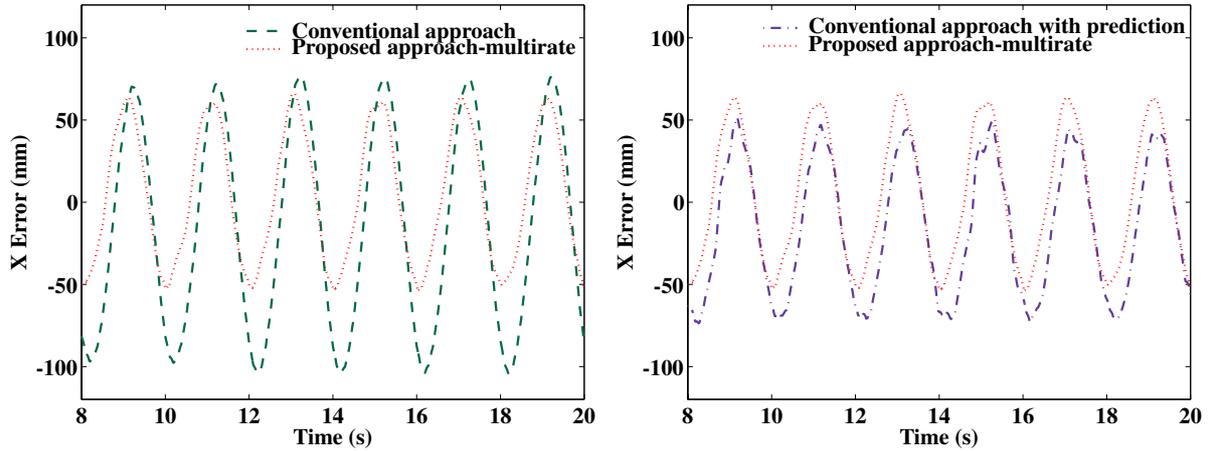


Figure 5.22: Tracking error for circular target motion(experiment)

hole with constant rotational velocity of  $\pi/3$ [rad/s] to show the square trajectory of the background. Fig. 5.23, Fig. 5.24 and Fig. 5.25 show that although the conventional approach with prediction can compensate the delay of vision sensor drastically, the accuracy of tracking trajectory is sacrificed because it enlarges the measurement noise caused from quantization error. Moreover, Fig. 5.23, Fig. 5.24 and Fig. 5.25 also show that the drastic effectiveness of our proposed intersample predictor control scheme. It can compensate the delay of vision sensor without sacrificing the accuracy of tracking trajectory.

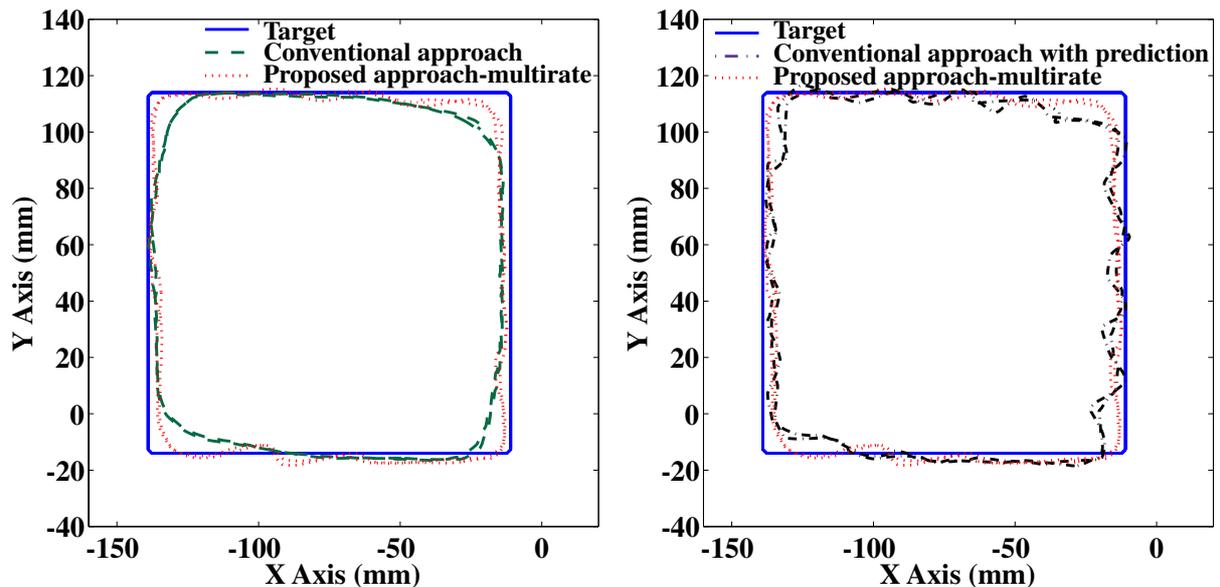


Figure 5.23: Tracking trajectory for square target motion(experiment)

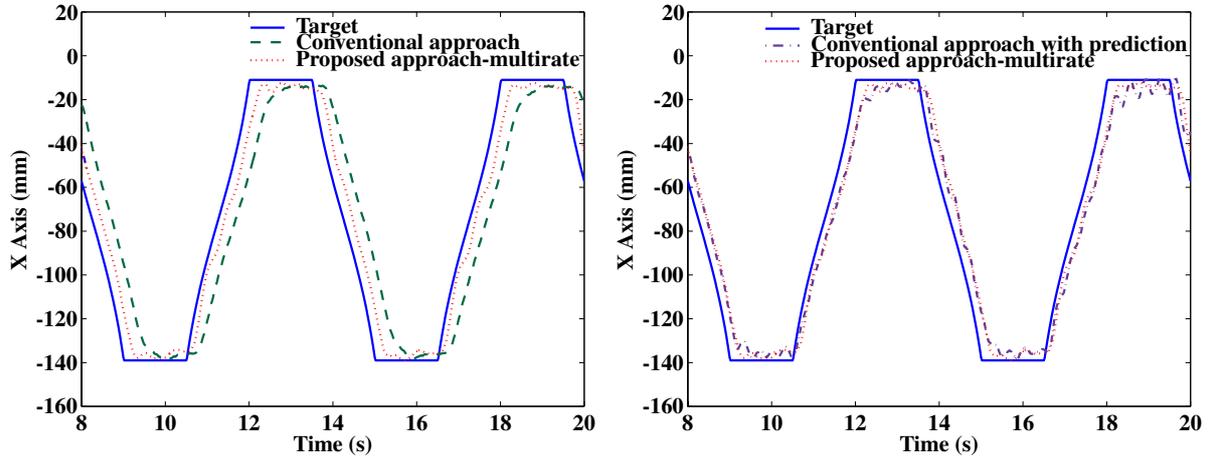


Figure 5.24: Tracking response for square target motion(experiment)

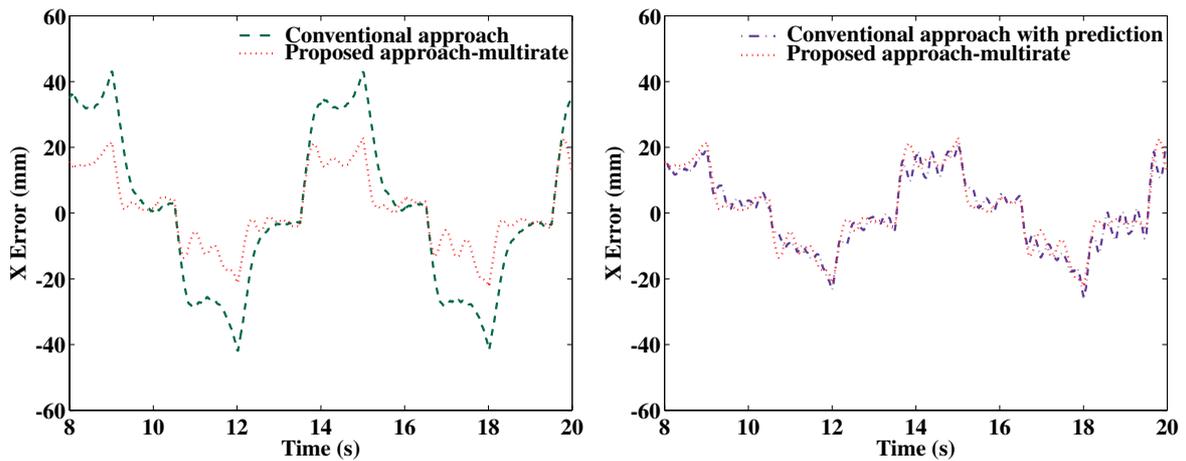


Figure 5.25: Tracking error for square target motion(experiment)

## 5.4 Summary

In this chapter, computer simulations and experiments using a 2-link DD robot were carried out to examine the effectiveness of the proposed methods. The results showed that the conventional approach has long time delay and worse control accuracy. Although the results showed that the conventional approach with prediction could compensate the delay of vision sensor drastically, the accuracy of tracking trajectory was sacrificed because it enlarges the measurement noise caused from quantization error. Moreover, the actual drastic effectiveness of our proposed intersample predictor control scheme based on multirate GPC was confirmed from both simulation results and experimental results. It could compensate the delay of vision sensor without sacrificing the accuracy of tracking trajectory.



# Chapter 6

## Conclusion

### 6.1 Summary

In this paper, focus on the most outstanding merit of machine vision, to make robot be able to operate autonomously in uncertain environment. Thus, instead of preview control, prediction control was discussed. There exists several problems when machine vision is used as feedback sensor. The well-known and remarkable problem is the relatively low sampling rate. That is to say, the visual system exhibits a long time delay. In preview control, it is relatively easy to compensate this delay by feedforwarding future values. In visual servoing, since the target motion is unknown, prediction is needed to eliminate time delay.

In this paper, we explained the cause of coordinate transformation problem is due to the multirate sampling characteristics of visual servo system from a viewpoint of prediction control while performing a high-speed tracking. Simulations were carried out to examine that this problem occurred in the conventional approaches (single-rate and multirate, without prediction and with prediction). Then two novel approaches, intersample prediction control schemes based on multirate GPC for high-speed visual tracking tasks, were presented. The fastrate GPC controller had good control performance while the measurement noise was small. In this work, we used a web camera as the machine vision in place of expensive machine system, so measurement noise due to quantization error become larger. Fastrate controller was not appropriate. However, another proposed approach, multirate controller, could solve this problem. The drastic performances of proposed approach, higher control accuracy and lower phase delay, were demonstrated through computer simulations and experiments using two-link direct drive robot and by comparing the conventional control approach and the conventional prediction control approach.

## 6.2 Problems and Future Works

### 6.2.1 Timing Problem of RTLinux

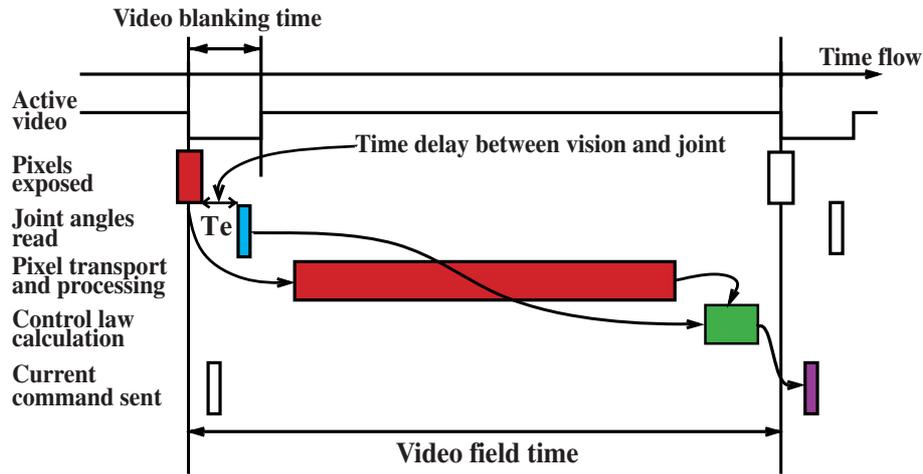


Figure 6.1: Details of single-rate system timing

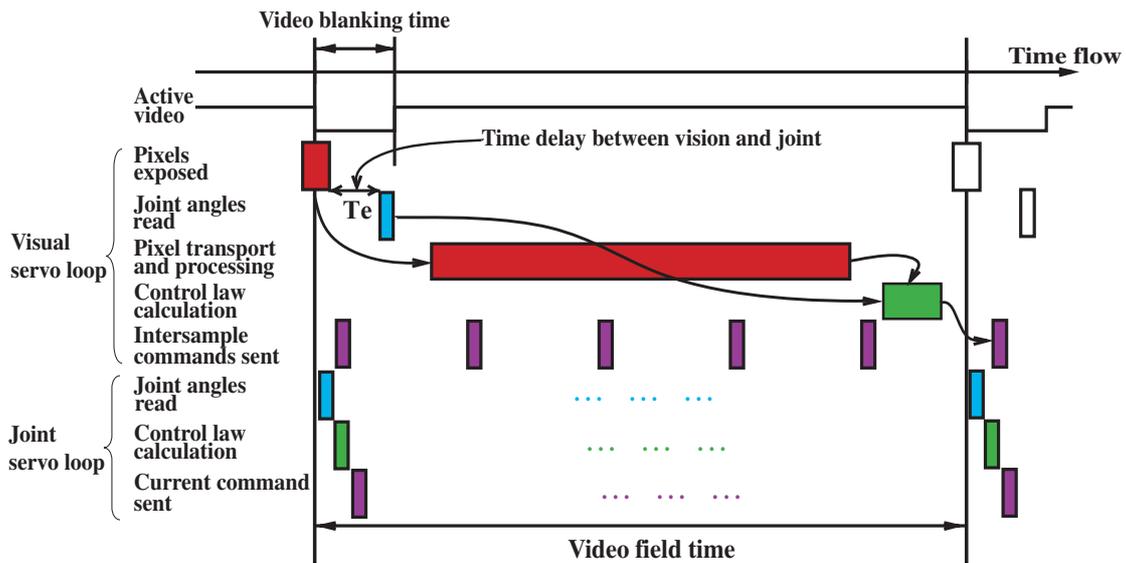


Figure 6.2: Details of multirate system timing

Fig. 6.1(From[3]) shows details of the important timing relationships involved. The actual instant of camera sampling is not generate by the camera. A short exposure time is required in order for the camera to approximate an ideal sampler of the visual state which is assumed for the control design. The robot's joint angles are sampled by a task during

the vertical blanking interval and combined with the image plane centroid to estimate the target position.

Fig. 6.2 shows details of the important timing relationships involved in 2 feedback loops of visual servoing. In single-rate system, all tasks can be scheduled by one direct flow. In multirate system, it is impossible to take several tasks at the same time. Real Time Linux must be used to solve this problem. From Fig. 6.2, we can see there exists a time delay between pixels exposed and joint angles read. It is quite a serious problem both in single-rate and multirate systems based on position-based control. The equation used to estimate the target position is given by

$$\mathbf{s}_o[k] = \mathbf{s}_c[k] + \mathbf{s}_e[k] \quad (6.1)$$

where  $\mathbf{s}_o$  is target position,  $\mathbf{s}_c$  is the camera position(the end-effector of robot),  $\mathbf{s}_e$  is the position difference between target and camera.

Because of the time delay between pixels exposed and joint angles read, (6.1) is rewritten as

$$\mathbf{s}_o[k] = \mathbf{s}_c[k + Te] + \mathbf{s}_e[k] \quad (6.2)$$

As a result, the estimation of target position will be inaccurate especially when  $Te$  cannot be neglected. It is necessary for our further task to eliminate it by determining the delay experimentally with respect to the video waveform and using the RTLinux program.

## 6.2.2 High-Ordered Predictive Model

Some predictive methods were mentioned in 4.3. Their common point is the need for a precise predictive model. When the dynamics of target motion is known, analyzing the dynamics to establish a predictive model is possible. However it is hard to decide a predictive model if target motion is unknown. We know any function(target trajectory) can be presented by Taylor series. In 4.3 we only used third order predictive model for target state estimation. By definition, the higher order the predictive model is, the higher precision it will have. Therefore, to use a high-order predictive model(see equation(6.3)) based on Taylor series is a possible method to establish precise predictive model for unknown target motion. However the effects of noise will become serious in high-ordered predictive model due to the calculus of finite differences. Incorporation of a phase-compensated parameter,  $\alpha$ , into predictive Kalman filter and finally forming an intersample predictor using polynomial interpolation(see Fig. 6.3) could eliminate noise drastically. Here we examine it by several simulations(see Fig. 6.4).

The high-ordered state-transition matrix with  $\alpha$ ,  $\mathbf{F}(\alpha)$  and the observation matrix,  $\mathbf{H}$  are given by

$$\mathbf{F} = \begin{bmatrix} 1 & \alpha T_f & \frac{(\alpha T_f)^2}{2!} & \frac{(\alpha T_f)^3}{3!} & \frac{(\alpha T_f)^4}{4!} \\ 0 & 1 & \alpha T_f & \frac{(\alpha T_f)^2}{2!} & \frac{(\alpha T_f)^3}{3!} \\ 0 & 0 & 1 & \alpha T_f & \frac{(\alpha T_f)^2}{2!} \\ 0 & 0 & 0 & 1 & \alpha T_f \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{H} = [1 \ 0 \ 0 \ 0 \ 0] \quad (6.3)$$

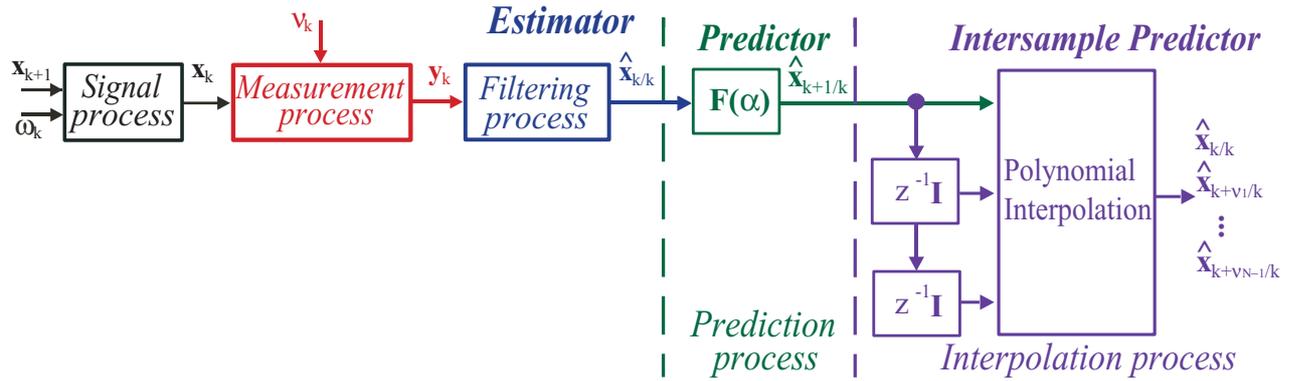


Figure 6.3: Predictive model incorporated a phase-compensated parameter

where  $T_f$  is the sampling period of vision sensor.

Fig. 6.4 shows the trend of tuning  $\mathbf{R}$  and  $\alpha$ . The sum of  $\|e\|^2$  can be lowered to hundredth of its original value due to tuning  $\alpha$ . Using this predictor with fastrate controller can perform a high performance visual servo system in the further tasks.

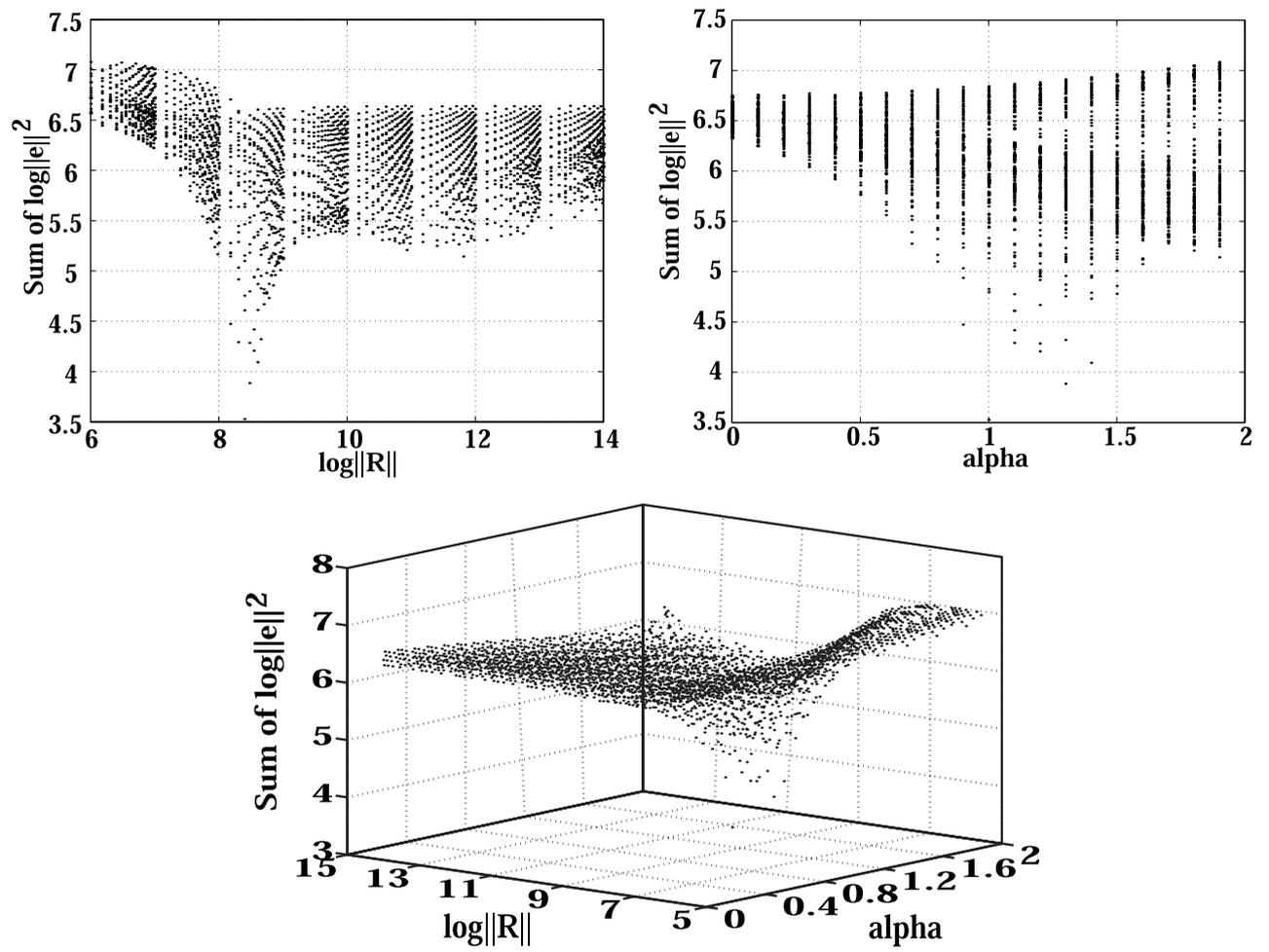


Figure 6.4: Prediction error while tuning  $K$  and  $\alpha$



# Appendix A

## The Experimental Setups of Machine Vision

Machine vision is the key channel that a visual servo system used to connect with the surrounding. The performance of machine vision is obviously influenced by the methods used to set up it. In this section, a detailed explain is given according to the machine vision we used.

Table A.1: Softwares used for visual servoing

Name	Version
Red Hat Linux	7.2
RTLinux Kernel	2.2.19
RTLinux	3.1
I2C	2.7.0
BTTV	0.7.80

Table A.2: Elements of machine vision

Hardware Name	Type
CCD Camera	CK-200
Capture Board	GV-VCP/PCI

Table A.1 shows the softwares used for visual servoing. The OS of the workstation is Red Hat Linux 7.2. RTLinux Kernel 2.2.19 is used for RTLinux 3.1 to build and compile with. The file system used to partition HDD is ext2 because ext3 is not supported. I2C is used to enable I2C Bus for BTTV. BTTV is a linux driver for TV cards based on the bt848 and bt878 chips. The image processing chip in our capture board is bt878 (shown in table A.2).

The calibration method of visual sensor was introduced in 2.2.1. We didn't worry too much about distortion since we were servoing mostly to a single point, bringing it to the center where there is very little distortion. The radial distortion could be neglected in our work. From viewpoint of calculation, according to the specification of CCD Camera CK-200 and a verified experiment, the projection matrix  $\mathbf{G}$  was given by

$$\mathbf{G} = \begin{bmatrix} 0.2152 & 0 \\ 0 & 0.1582 \end{bmatrix} \quad (\text{A.1})$$

# Appendix B

## The Software for Visual Servoing

Fig. B.1 shows the real time tasks and FIFO interlinked diagram of the software for visual servoing. Two threads are created and 6 RT-FIFO devices are opened. Thread 1 is for the control calculation of joint servo loop. Thread 2 is for the control calculation of visual servo loop. Rtf3, rtf4 and rtf5 are input control channels. Rtf1 is a output data channel for thread 1 to send data from RT-kernel space to user space. Rtf2 and rtf3 are input data channels for user space to send data(extracted feature and angles) to RT-kernel space.

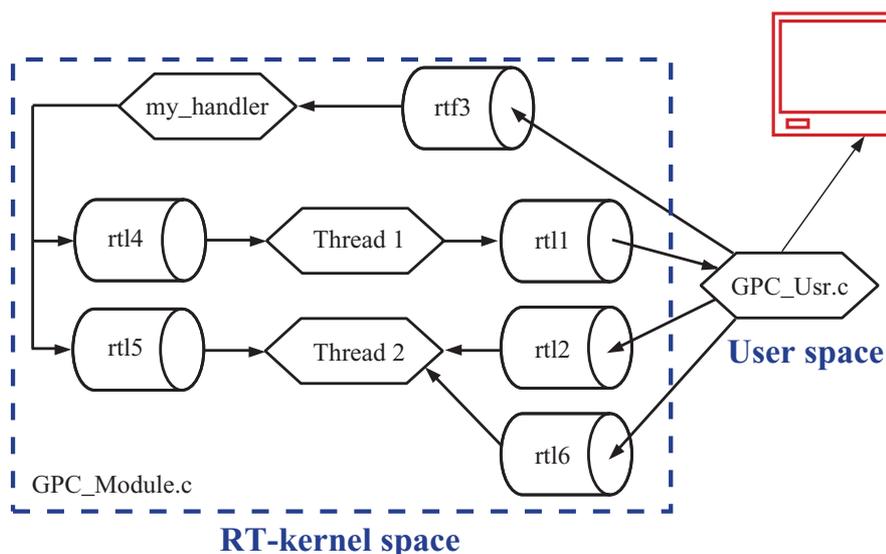


Figure B.1: Real time task create, FIFO interlinked diagram

Fig. B.2 shows the actions of camera. This process is carried out in user space. In the step of Initial Video, there are five ioctl[42] commands are performed: get the capabilities of the framegrabber, get the information from the input channel, select the video input channel, set the properties of the video stream and get information of the video buffers of

the driver. The step, Capture Image, starts a capture to one of the video buffers(frames) of the driver. The following step, Read Angles is used to obtain the right timing angles for estimating target position. In the step, Wait Capture, waits for a grab started with Capture Image to be completed. In last step, Feature Extraction, obtains the position of the target in camera coordinate system and sends it to thread 2 in RT-kernel space for calculating the visual servo loop control inputs.

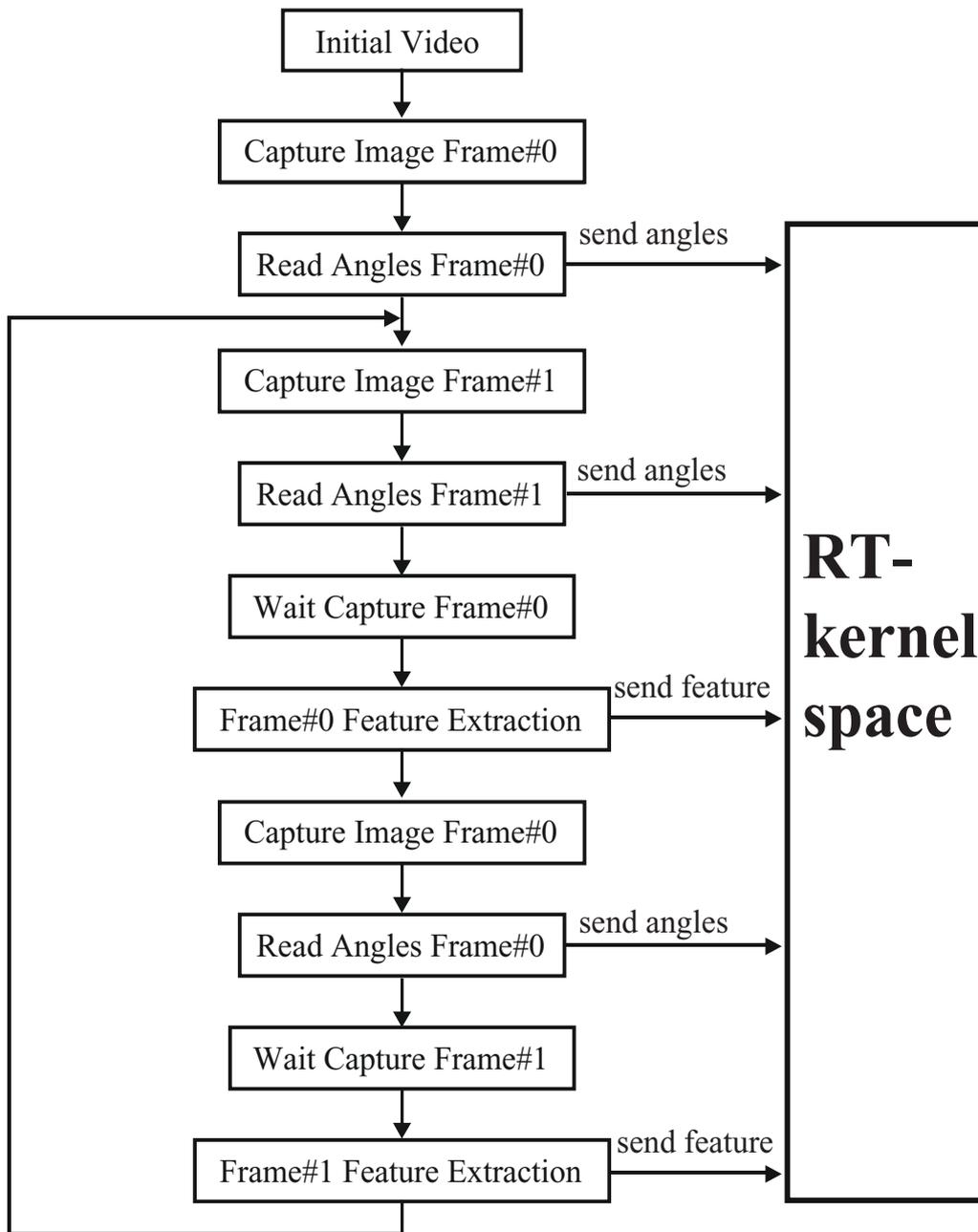


Figure B.2: Camera action flow chart

# Bibliography

- [1] “RTS PixCell<sup>TM</sup>”,  
<http://www.automaatioseura.fi/jaostot/mvn/mvn6/goods.html>.
- [2] “MIT’s Intelligent Transportation Research”,  
<http://www-mtl.mit.edu/mtlhome/6Res/ITRC/itrc.html>.
- [3] P.I. Corke: “Visual Control of Robots: high performance visual servoing”, Research Studies Press, 1996.
- [4] K. Hashimoto and T. Noritsugu: “Observer-based control for visual servoing”, In 13th IFAC World Congress, Vol.F pp.453-458, San Francisco, 1996.
- [5] J.A. Gangloff, M.de Mathelin and G. Abba: “6 DOF high speed dynamic visual servoing using GPC controllers”, In IEEE Int. Conf. Robotics and Automation pp.2008-2013, 1998.
- [6] J.A. Gangloff and M.de Mathelin: “High speed visual servoing of a 6 DOF manipulator using MIMO predictive control”, In IEEE Int. Conf. Robotics and Automation pp.3751-3756, 2000.
- [7] J.A. Piepmeier, G.V. McMurray and H. Lipkin: “Tracking a Moving Target with Model Independent Visual Servoing: a Predictive Estimation Approach”, In IEEE Int. Conf. Robotics and Automation pp.2652-2657, 1998.
- [8] B. Espiau: “Effect of camera calibration errors on visual servoing in robotics”, In Third Int. Symposium on Experimental Robotics, Kyoto, Japan, October 1993.
- [9] F. Chaumette: “Potential problems of stability and convergence in image-based and position-based vervo servoing”, in The Confluence of Vision and Control, D. Kriegman, G . Hager, A.S. Morse (eds.), Pages 66-78, LNCIS Series, No 237, Springer-Verlag, 1998.

- [10] R. Y. Tsai: “A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses”, *IEEE J. Robotics Automat.*, pages 323-344, Vol. RA-3, No. 4, 1987.
- [11] T. P. Sim, G. S. Hong and K. B. Lim: “Multirate predictor control scheme for visual servo control”, *IEE Proc.-Control Theory Appl.*, Vol. 149, No. 2, March 2002.
- [12] H. Fujimoto, Y. Hori: “High Performance Servo Systems Based on Multirate Sampling Control”, *IFAC Journal on Control Engineering Practice*, vol. 10, no. 7, pp. 773–781, 2002.
- [13] K. V. Ling and K. W. Lim: “A State Space GPC with Extensions to Multirate Control”, *Automatica*, Vol. 32, No. 7. pp. 1067–1071, 1996.
- [14] Y. Nakabo, M. Ishikawa, H. Toyoda and S. Mizuno: “1ms Column Parallel Vision System and Its Application of High Speed Target Tracking”, *IEEE Int. Conf. Robotics and Automation*, pp.650–655, 2000.
- [15] S. H. Nam and S. Y. Oh: “Real-Time Dynamic Visual Tracking for Robot Manipulators Using Position Sensitive Detectors, *Applied Intelligence*”, Vol. 10, No. 1, pp. 53–70, 1999.
- [16] R. L. Andersson: “Understanding and applying a robot ping-pong player’s expert controller”, *Proceedings of the IEEE International Conference on Robotics and Automation*, 1284–1289, 1989.
- [17] D. W. Clarke, C. Mohtadi, and P. S. Tuffs: “Generalized predictive control - part I, the basic algorithm”, *Automatica*, 23:137–160, 1987.
- [18] S. Masuda, A. Yanou, A. Inoue and Y. Hirashima: “Construction Method of Generalized Predictive Control System in the State Space Approach Which is Equivalent to One in the Polynomial Approach and a Proof of the Equivalence”, *Trans. SICE*, vol. 35, no. 2, pp. 221–230, 1999. (in Japanese)
- [19] T. Hagiwara and M. Araki: “Design of a state feedback controller based on the multirate sampling of plant output”, *IEEE Trans. Automat. Contr.*, 33, 9, pp. 812–819, 1988.
- [20] T. Chen and B. Francis: “*Optimal Sampled-Data Control Systems*”, Springer, 1995.
- [21] K. Hashimoto and H. Kimura: “Visual servoing with nonlinear observer”, *IEEE Int. Conf. Robotics and Automation*, pp. 484–489, 1995.

- [22] M. Nemani, T. C. Tsao and S. Hutchinson: “Multi-rate analysis and design of visual feedback digital servo-control system”, ASME, J. Dynam. Syst., Measur., and Contr., 116, pp. 44–55, 1994.
- [23] J. T. Feeddma and O. R. Mitchell: “Vision guided servoing with feature-based trajectory generation”, IEEE Trans. Robotics and Automation, 5, 5, pp. 691–700, 1989.
- [24] Y. Hori and K. Ohnishi: “Applied control engineering”, Maruzen, 1988. (in Japanese).
- [25] T. Mita, S. Hara and R. Kondo: “Introduction to Digital Control”, CORONA Publishing Co., LTD., 1988. (in Japanese).
- [26] H. Fujimoto and Y. Hori: “Visual servoing based on multirate sampling control”, IEEE Int. Conf. Robotics and Automation, 2001.
- [27] “FSMLabs - The RTLinux Company”,  
<http://www.rtlinux.org/>.
- [28] L. E. Weiss, A. C. Sanderson and C. P. Newman: “Dynamic sensor based control of robots with visual feedback”, IEEE J. Robotics and Automation, 3, 5, pp. 404–417, 1987.
- [29] S. Hutchinson, G. D. Hager and P. I. Corke: “A tutorial on visual servo control”, IEEE Trans. Robotics and Automation, 12, 5, pp. 3745–3750, 1996.
- [30] T. Murakami and K. Ohnishi: “A study of stability and workspace decoupling control based on robust control in multi-degrees-of-freedom robot”, Trans. IEE of Japan, 113-D, 5, pp. 639–646, 1993. (in Japanese).
- [31] K. Hashimoto, T. Ebine and H. Kimura: “Visual servoing with hand-eye manipulator optimal control approach”, IEEE Trans. Robotics and Automation, 12, 5, pp. 766–774, 1996.
- [32] M. Nakadokoro, S. Komada and T. Hori: “Stereo visual servo of robot manipulators by estimated image feature without 3D reconstruction”, IEE of Japan, JIASC '99, Vol. 1, pp. 451–454, 1999. (in Japanese).
- [33] D. B. Zhang, L. V. Gool and A. Oosterlinck: “Stochastic predictive control of robot tracking system with dynamic visual feedback”, Proc. IEEE Int. Conf. Robotics and Automation, pp. 610–615, 1990.

- [34] C. Smith and N. Papanikolopoulos: "Vision-guided robotic grasping: issues and experiments," Proc. IEEE Int. Conf. Robotics and Automation, pp. 3203-3208, 1996.
- [35] P.I. Corke and M.C. Good: "Dynamic effects in visual closed-loop system," IEEE Trans. Robotics and Automation, 12, 5, pp. 671-683, 1996.
- [36] P.I. Corke: "Dynamic issues in robot visual-servo systems," In G. Giralt and G. Hirzinger, editors, Robotics Research. The Seventh International Symposium, pages 488-498. Springer-Verlag, 1996.
- [37] T. Murakami, N. Oda, Y. Miyasaki and K. Ohnishi: "A motion control strategy based on equivalent mass matrix in multidegree-of-freedom manipulator," IEEE Trans. Industrial Electronics, 42, 5, pp. 123-130, 1995.
- [38] K. Saruda, H. Fujimoto and Y. Hori: "Visual servoing system with feature prediction using motion observer," Trans. IEE of Japan vol.122-D, no.5, pp. 516-521, 2002. (in Japanese).
- [39] R.E. Kalman: "A new approach to linear filtering and prediction problems," Trans. ASME, J Basic Engrg., vol. 82, pp. 34-45, March 1960.
- [40] Y. Hori: "High Performance Control of Robot Manipulator without using Inverse Dynamics," IFAC Control Engineering Practice, Vol.1, No.3, pp. 529-538, 1993.
- [41] P.I. Corke and M.C. Good: "Controller design for high speed-performance visual servoing," In 12th IFAC World Congress, Vol.9 pp. 395-398, Sydney, Australia, 1993.
- [42] "Video4Linux API",  
<http://gate.csl.mtu.edu/lxr/http/source/Documentation/video4linux/API.html>.

# List of Publications

## International conference

1. JiunDe Wu and Yoichi Hori, “An Intersample Predictor Control Scheme Based on Multirate GPC for High-Speed Tracking Tasks” in The 29th Annual Conference of the IEEE Industrial Electronics Society (IECON03), Roanoke(Virginia), 2003.(to be presented)

## Domestic conference

1. JiunDe Wu and Yoichi Hori, “Proposal of High Speed Visual Servo System Focused to Solve the Interference Problem of Coordinate Transformation Using Intersample Predictor,” IEE of Japan Technical Meeting Record, no. IIC-01-13, pp. 107–112, 2003.(in Japanese)
2. JiunDe Wu and Yoichi Hori, “Novel Design of Intersample Predictor Based on Multi-rate GPC for High-Speed Visual Servoing,” 2004 National Convention Record I.E.E Japan – Industry Application Society –, Vol. 2, no. 2-55, pp. 565–568, 2003.

## Journals

1. Plan to submit a fullpaper to IEEE Transaction Journals.