Multiate Digital Control for Robotic Visual Servoing Tasks

by

Muhammad Sarmad Ali

A thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science

The Department of Computer Science

National University of Computer and Emerging Sciences
Lahore, 54700
Pakistan

September, 2004

Approved: _____

Head
Department of Computer Science
National University of Computer and Emerging Sciences
Block – B, Faisal Town
Lahore, Pakistan

_____20 ____

Vita

Mr. Muhammad Sarmad Ali was born in Sargodha, Pakistan on March 31, 1977. After doing F.Sc. (Pre-engineering) from Government College, Sargodha in 1994, he joined University of Engineering and Technology (UET), Lahore. He received a Bachelor of Science in Mechanical Engineering from UET, Lahore in year 2000. He started M.Sc. Computer Science from National University of Computer and Emerging Sciences, Lahore, in year 2001. The research in this thesis report was carried out from 2003 to 2004.

He enjoys reading Islamic history and urdu literature, and playing table tennis.

Dedication

To my Parents,

Sister, and Brothers

Abstract

The objective of this thesis is to investigate the use of *multirate digital control* in visually servoing a robot manipulator for position control. *Visual servoing* is the application of computer vision for closed-loop control of a robot manipulator, and has the potential to solve a number of problems that currently limit the potential of robots in advanced applications.

Robot visual servo systems are inherently multirate systems since different sampling rates are present. Though, computer vision systems have been used for robot control for over past two decades, and multirate control has also been well studied during that period, still quite a few efforts emerged to exploit the combination of vision and multirate control theory for visual servoing. A popular approach has been to approximate the multirate system by a single-rate system, in order to utilize conventional single-rate control techniques.

This thesis applies a multirate digital control strategy to a 2D visual servoing problem. This control strategy has successful application in hard disk drive control. Since visual tracking is an integral part of nearly all visual servoing tasks, this thesis tries to explore the usefulness of multirate scheme for visual tracking. The effectiveness of multirate control approach is verified by simulation considering a planar 2DOF rigid-link robot manipulator with a camera-mounted end-effector, actively tracking a moving object. A simple single-rate control scheme is also simulated for comparison. It is observed that multirate-based visual servoing results in better performance while trajectory following.

Acknowledgements

All the praise and thanks to ALLAH, the creator. Only HE possesses the ultimate knowledge. HE provides the will and courage to delve into the deep sea of knowledge.

Salam on Prophet MOHUMMED (peace be upon him), who is the ‘City of Knowledge’. He guided us through the path, and motivated us to seek knowledge.

My sincerest thanks to Dr. Syed Amjad Hussain Shah whose guidance and support always encouraged me during the whole tenure of the thesis work. He has been kind enough to provide me the opportunity to work as a Research Officer with him for one year. He actively reviewed the research work and papers, and provided useful suggestions. I am much grateful to Dr. Mian Muhammad Awais for his guidance and encouraging me to present my first paper at IBCAST conference. I offer thanks to Dr. Muhammad Akaml Butt for his directions and useful comments about simulation. I am also obliged to Dr. Arshad Hussain for his support and directions.

Finally, I offer my gratitude to my family, and my friends Hammad Sharif, Kashif Saeed and Asad Riaz for their support and prayers for my success. Special thanks to Hammad Sharif for his help and useful discussions.

Contents

List of Symbols

$\triangleq$	Defined as
r	Reference signal
e	Error signal
u	Input signal
y	Output signal
$\hat{y}$	Approximate or estimate of signal y
w	Process noise/disturbance
v	Measurement noise/disturbance
$\mathbf{a}$	Row or column vector
$\mathbf{A}$	Matrix
$\mathbf{A}^T$	Transpose of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$
$\mathbf{I}$	Identity matrix
$\mathbf{O}$	Zero matrix
$G(s)$	Continuous-time system
$G[z]$	Discrete-time system
$x(k)$	Continuous-time signal
$x[k]$	Discrete-time signal
$\left[\begin{array}{c\|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right]$	$\triangleq \mathbf{C}(z\mathbf{I}-\mathbf{A})^{-1}\mathbf{B}+\mathbf{D}$ (Short-hand for state-space realization)
L	Lifting operator
ZOH	Zero-Order-Hold
$\Re$	Set of real numbers
$\|..\|_2$	2 – Norm
$\|..\|_\infty$	$\infty-$Norm

List of Abbreviations

A/D	Analog to Digital Converter
BIBO	Bounded-Input Bounded-Output
CCD	Charge-Coupled Device
D – H	Denavit – Hartenberg representation
D/A	Digital to Analog Converter
DOF	Degrees of Freedom
FDLTI	Finite Dimensional Linear Time Invariant
FOV	Field of View
HDD	Hard Disk Drive
IBVS	Image-Based Visual Servoing
LQG	Linear Quadratic Gaussian
LTI	Linear Time-Invariant
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
MRSDS	Multi-Rate Sampled-Data Systems
PBVS	Position-Based Visual Servoing
SISO	Single-Input Single-Output
ZOH	Zero-Order Hold

List of Figures

Chapter 1

Introduction

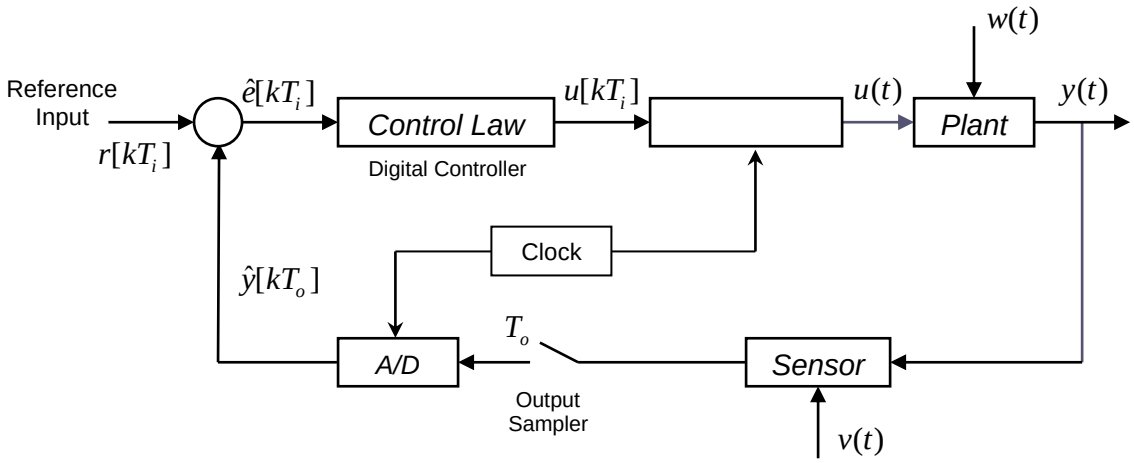
Owing to recent developments of the computer and interfacing hardware, digital controllers are utilized for controlling almost all electro-mechanical systems such as robots, motors, machine tools, hard disk drives etc. because of reliability and compactness. Among the advantages of digital approaches for control is the increased flexibility of the control algorithms, which allows combining the same control logic with proper system dynamic functions to meet the needs of various types of systems. Advanced computation power availability has also tremendously improved the time and cost effectiveness towards modeling, analysis and design of control systems.

Digital control offers techniques that are not offered by continuous time design techniques. Finite time control or *dead-beat* control is an example. However, sampling of continuous time signals implies loss of some information, and some unexpected problems can appear in the intersample periods. *Multirate* digital control techniques which update the control input faster than the output measurements can be used to overcome this problem [1], [2], [3], [5], [4], [6].

The block diagram of a general single-input single output (SISO) multirate control system is shown in Figure 1-1. The plant is continuous while the controller is discrete. In order to implement a digital controller, analog-to-digital (A/D) and digital-to-analog (D/A) converters are needed. In a simple single-rate system, A/D and D/A converters are synchronized by a clock such that the *sampling*[1] and *hold*[2] rates are equal i.e. $T_i = T_o$.

[1] From now on, *sampling* will implicitly imply that there is some A/D conversion mechanism

Situations most likely arise in practical systems where this equality cannot hold. The control input time period T_i and the measurement output time period T_o are unequal. Such systems that have two or more sampling or hold elements operating at different frequencies are generally known as *multirate* systems. The system shown in Figure 1-1 is a multirate system.



r = reference or command inputs
u = control or actuator input signal
y = controlled or output signal
$\hat{y}$ = instrument or sensor output usually an approximation to or estimate of y
$\hat{e} = r - \hat{y}$ = indicated error
$e = r - y$ = system error
w = distrubance input to the plant
v = distrubance or noise in the sensor
A/D = analog-to-digital converter
D/A = digital-to-analog converter

Figure 1-1: A simple SISO multirate control system

1.1 Background

It is well known in digital control that the sampling rate is a critical design parameter. Many system properties, such as tracking performance, rejection of plant disturbances and measurement noise, sensitivity to unmodeled dynamics etc. are influenced by the sampling rate. It is widely accepted that a better control performance can result if a fast sampling rate is selected relative to the desired closed loop bandwidth [3]. However the achievable sampling rate is usually restricted since most of the real world systems are constrained to have different sampling rates at different sampling and hold devices due to:

[2] From now on, *hold* will implicitly imply that there is some D/A conversion mechanism. When refereeing to hold operation, a zero-order hold will be assumed, unless stated otherwise.

- Hardware restrictions
- Computing time limitations
- Speed of the actuator
- Speed of the sensor
- System dynamics

Typical examples of such systems are many process control applications where composition measurements in distillation columns are typically sampled at a slower rate than temperatures and flow rates [7]. High track density HDD systems are a classical example of multirate electro-mechanical systems. Here sector-embedded servo signal scheme limits the output sampling rate. In complex, multi-variable control systems, it is often unrealistic, and sometimes impossible, to sample all physical signals uniformly at one single rate. This creates the need for a multirate control strategy.

1.2 Motivation

Vision-based systems are another typical example where sampling rates are constrained. Vision based motion control systems e.g. robot visual servo systems are inherently multirate systems. Visual feedback loop clearly have much slow update rate since the camera frame rate is quite limited as compared to other sensors. Robot joint and actuator sensors can provide feedback information at a much higher rate than the visual feedback by an outer camera loop. This observation provides primary motivation for this work.

Early visual motion control approach, the *static look-then-move* approach, alternated the systems between image capturing and decision making [8], [9]. This phenomenon substantially degrades the performance of control structure specifically and overall system in general. With today's fast-processing technology, time interval between two successive images is sufficient to extract meaningful information for control calculation. It has been proposed that a visual motion control strategy which sends control commands to the plant N times ($N > 1$) while incorporating slow rate visual loop in control law can result in smooth, high performance motion control [10].

A major source of motivation for this work comes from Fujimoto et al. [10]. They successfully addressed the issue using multirate sampling control, proposing a novel design approach for perfect disturbance rejection and tracking control. Though, multirate nature of visual servo systems was known since their very early realizations [8], there has

been little effort to incorporate multirate digital control schemes. Quite a little research material can be found on multirate design of visual servo systems.

However, multirate digital control has successfully been applied to the classical problem of disk drive control (see e.g. Lee et al. [11] and references cited therein). The selection of the design technique is inspired by their work. They have proposed a fixed order multirate control design approach. They have declared improvements in tracking control of disk drive actuator arm, without exciting flexible mode dynamics of the arm and suspension. In addition, their approach is much simpler, more computation effective and easy to implement. We have tried to investigate this multirate control design technique for visual servoing of robot manipulators. Investigation is based upon simulation of 2DOF planar robot with end-effector mounted camera.

1.3 Outline of the Thesis

Outline of the thesis is as follows:

Chapter 2 presents a brief theoretical perspective on multirate digital control systems. A short note on historical development of multirate systems is given. Overview of *successive loop closure* technique and direct design through *lifting* is presented.

Chapter 3 provides a brief account of theoretical background on visual servoing. Major visual servoing architectures are discussed. A control theoretic perspective is presented.

Chapter 4 starts with the problem description and presents a short overview of previous efforts towards multirate design of visual servo systems. Proposed multirate control approach is described in a bit detail.

Chapter 5 describes the simulation setup for the thesis work. Assumptions are stated and parameters are defined.

Chapter 6 discusses the results obtained by simulation using aforementioned setup. Comparison between a simple single-rate scheme and designed multirate scheme is made.

Chapter 7 finally concludes the thesis with possible directions for future work.

Chapter 2

Multirate Digital Control

A SISO system is presented in previous chapter. Vast majority of practical systems to be controlled are, however, MIMO systems. MIMO multirate systems represent the general class of multirate systems. For conventional digital control design of such systems, it is assumed that they are single-rate MIMO systems, where both the control inputs and the plant outputs are updated synchronously at same rate. However, the single rate approach is usually not the best approach from technical and/or economical perspective [3], [5]. For MIMO systems to obtain the best possible performance, it is often necessary to sample the signals for the various sensors and control effectors at different rates. However, unfortunately, control system design methods are well developed and practiced only for the special case where all sampling rates are the same.

From sampling-rate perspective, multirate systems can be classified into *regular* and *irregular* multirate systems. It is generally accepted that there is a *sampling periodicity* for each signal, and there exists a least common multiple period T_f, called the *frame period*, and a greatest common divisor T_b, called the *base period*. In regular multirate systems the sampling rate of a particular signal is constant, though different signals can have different rate. While in irregular multirate systems, a particular signal is arbitrarily sampled in irregular intervals, though this irregular sequence is repeated after each frame period. A general multirate sampling scheme for MIMO systems is shown in Figure 2-1. With $N = lcm(N_i, M_j)$, for $i = 1,.....,m$ and $j = 1,.....,n$, the frame period $T_f = N T_b$. The i^{th} control input rate is N_i times the base period, while j^{th} output measurement is obtained by sampling at M_j times the base period (N_i and M_j are integers).

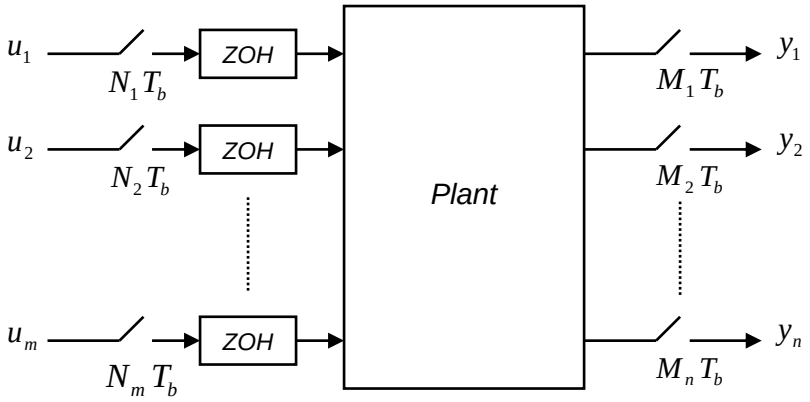


Figure 2-1: A MIMO multirate system

If the plant is a continuous-time process, the control problem is referred as a multirate *sampled-data* problem, whereas it will be a *discrete-time* multirate problem if the plant is assumed to be a discrete-time system. In most studies on multirate problems, the sampling times appearing in the system are assumed to be related through some rational fraction, so that the resulting multirate system will be periodic. In this thesis, the standard periodic multirate problem is considered.

Various multirate control problems presented in the literature can be classified into two types:

- *N-delay input control* when the input is updated *N* times the sample rate of the output
- *N-delay output control* when the output is sampled *N* times the update rate of the input.

In this thesis, the study of vision-based motion control under measurement constraints is discussed as a case of *N*-delay input control. Multi-rate controllers are, in general, time varying. Thus multi-rate control systems can outperform single rate systems [4], [5], [12].

2.1 Brief Historical Development

Multirate digital control is a noteworthy area of present research and application that is motivated by practical implementation needs. The drive for exploring multirate control techniques has traditionally been the aerospace and defense industry [13]. The field of

digital control and sampled-data[3] control originated from radar applications during World War-II [13]. Methods for effective design of control systems using sampled data were under initial development during the later 1940's. Interest in multirate systems theory began in the early 1950's as an analysis tool. It was possible to study the inter-sample behavior of a signal or output of a single rate control system by introducing a "phantom sampler" (i.e., a fictitious sampler that operates at a rate some integer ratio higher than that of the controller). A significant early contribution to this general method of analysis, known as *frequency decomposition,* was made by Sklansky and Ragazzini [14]. A similar frequency domain technique, known as *switch decomposition*, was developed by Kranc [15] as an extension of the frequency decomposition method. Theoretically, both methods can be used to determine transfer functions of many multirate systems of practical interest. In practice, however, these methods turnout to be extremely cumbersome [6].

A noteworthy treatment of multirate systems from time-domain perspective was presented by Kalman and Bertram [16] in 1959. Their work made a major contribution in showing the power and flexibility of *state-space* techniques. Quite a few efforts emerged for nearly next two decades.

A new move for research on multirate system theory started from early 1980's with an increased interest in multirate controller design. An unpublished bibliographic reference compiled by Mario Vélez [17] outlines nearly 400 published research papers related to multirate control from 1982 to year 2002. Optimal sampled-data control theory has been given a considerable attention for multirate problem. Since many industrial control systems consist of a continuous time plant and a discrete time controller interconnected via A/D and D/A converters, direct optimal control of multirate systems has been studied in the sampled-data setting. The multirate LQG problem has been treated in [6] (for further details, see references cited therein). The LQG method is advantageous due to the fact that it accounts for all cross coupling between control loops. However a drawback of the approach is that it always yields periodically time-varying compensators having the same dynamic order as the plant. The multirate H_2 and H_∞ control problems are treated in [1], [18]. Some of the more recent techniques apply *lifting* to the periodic multirate problem, which results in an LTI description with extended inputs and outputs [12].

[3] It is known that, in general, a periodic sampled-data multirate problem can be represented by an equivalent periodic discrete-time problem. Therefore no distinction is made between sampled-data and discrete-time multirate problems in this section.

2.2 Significant Design Approaches

Meyer et al. [19] showed that multirate systems are special cases of periodic systems. The *z*-transform method, which is the basis for the analysis of discrete systems, does not apply directly to multirate systems [3]. Special design and analysis techniques have been proposed for this purpose. A common fact in nearly all design approaches is to reduce or 'lift' a periodic multirate problem to a single-rate time-invariant problem. A multirate system can thus be represented by means of a linear time-invariant state equation. Several authors have shown that this representation preserves many of the properties of the original MR system like reachability, controllability, observability, BIBO stability etc. [19], [20]. Figure 2-2 below shows how different design approaches can be classified. A few significantly utilized techniques are briefly summarized below:

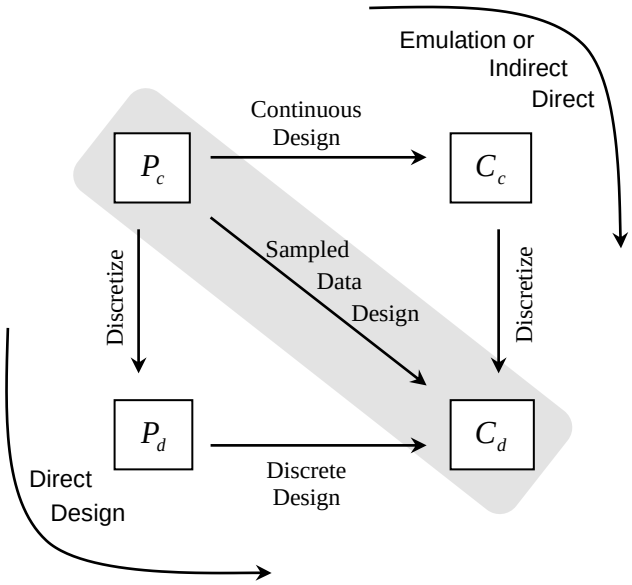


Figure 2-2: Design methodologies for discrete-time controller from continuous-time plant

2.2.1 Successive Loop Closure

The SLC method is an indirect approach to multirate synthesis. It is also an often used analysis method. The idea is to initially decouple the system into two or more single-rate subsystems as shown in the Figure 2-3. The controller for the subsystem with fastest sample rate is first designed in digital domain. This digitally controlled subsystem is then converted back to continuous system, allowing it to be meld into the slower rate subsystem. A discrete representation at the slow sample rate of the entire plant, including

the previously ignored cross-coupling and the fast controller, can now be obtained using conventional techniques. This model is exact provided the fast sample rate is an integer multiple of the slow rate.

A multirate compensator is thus synthesized by successively closing a series of single-rate control loops. The loops are closed in order according to sampling rate, from fastest rate to slowest rate. Suppose that the analog plant is represented by

$$\dot{\mathbf{x}}(t) = \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}_1\,\mathbf{u}_1(t) + \mathbf{B}_2\,\mathbf{u}_2(t) \qquad \textbf{(2-1)}$$

where $\mathbf{x}$ is the state, and $\mathbf{u}_1$ and $\mathbf{u}_2$ are the control inputs. Suppose that the $\mathbf{u}_2$ loop is closed first, at the sampling rate $1/T$, with zero order hold, using the discrete-time control law

$$\mathbf{u}_2[i] = -\mathbf{K}_2\,\mathbf{x}[i] \qquad \textbf{(2-2)}$$

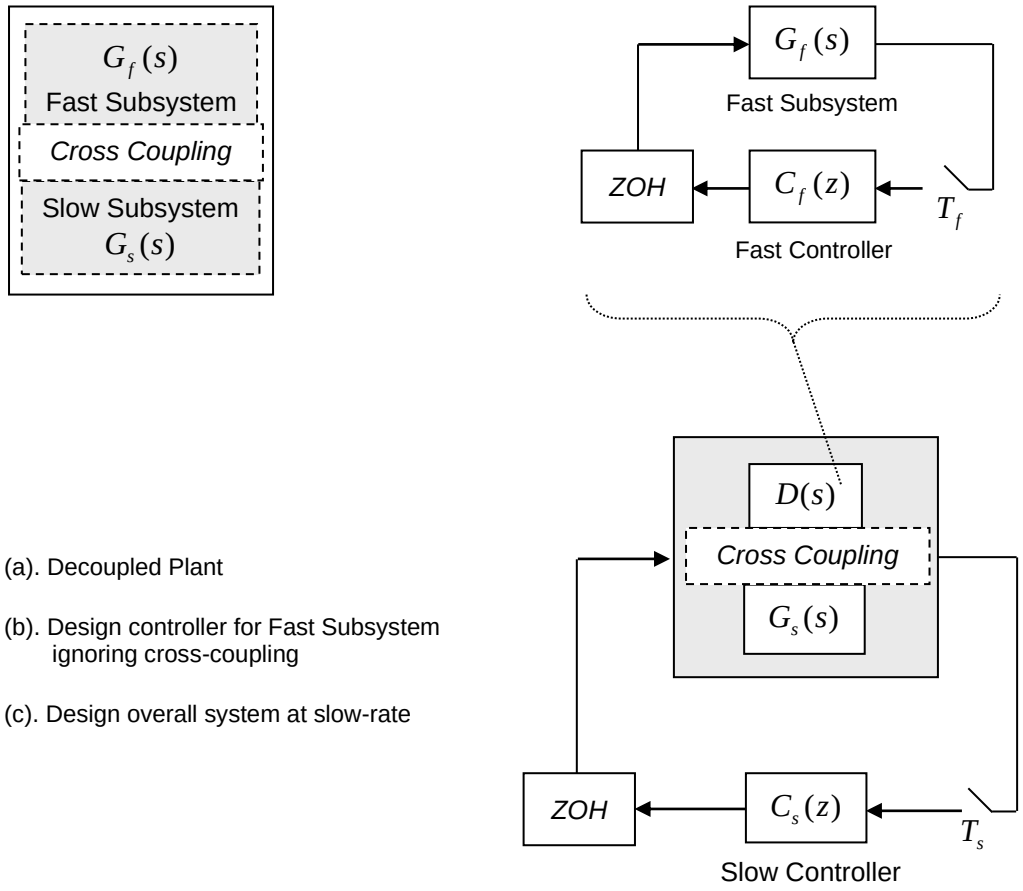


Figure 2-3: Simple SLC multirate design strategy

Suppose that the $\mathbf{u}_1$ loop is to operate at the slower sampling rate $1/mT$, where m is a positive integer. Then the plant model for the $\mathbf{u}_1$ loop design is

$$\mathbf{x}[i+1] = (\mathbf{\Phi} - \mathbf{\Gamma}_2\mathbf{K}_2)^m \mathbf{x}[i] + \mathbf{\Gamma}_1 \mathbf{u}_1[i] \qquad \textbf{(2-3)}$$

where
$$\mathbf{\Phi} \triangleq e^{\mathbf{A}T}$$
$$\mathbf{\Gamma}_1 \triangleq \int_0^{mT} e^{\mathbf{A}T} dt \,.\, \mathbf{B}_1$$
$$\mathbf{\Gamma}_2 \triangleq \int_0^{T} e^{\mathbf{A}T} dt \,.\, \mathbf{B}_2$$

There are two major advantages [3] of the SLC method; only conventional (typically SISO) single-rate synthesis techniques are required; and the resulting compensators usually have a simple and low-order structure. A disadvantage [6] of the SLC method is that it is an ad hoc method for dealing with the cross coupling between control loops. We can only design the n^{th} loop to respond to the cross-coupled effects of the $(n+1)^{\text{th}}$ loop as disturbances.

2.2.2 Direct Design through Lifting

Multi-rate plants do not have *z*-transform description since their input/output map is not shift-invariant. However, using ideas that date back to Kranc [15], it is possible to construct a shift-invariant equivalent, or *lift* of the multi-rate plant that have a *z*-transform description. The modern multirate system design is based on the technique called *lifting*, which facilitates simple design and analysis. It has been studied in detail by Khargonekar et al. in [21].

It is possible to lift both signals and systems. The idea of lifting in advanced sampled-data control theory has been discussed in [1]. A brief description of discrete time lifting is presented here.

Lifting Discrete-time Signals

Consider $\mathbf{V}$ as complex space of sequences of values defined over the set of nonnegative integers. We define the linear transformation L as $L : \mathbf{V} \to \mathbf{V}^m$. Under this transformation, given a discrete-time signal $\mathbf{u} = \{u(0), u(1), u(2), \ldots\} \in \mathbf{V}$ that refers to the period h, the signal

$$\underline{\mathbf{u}} = \left\{ \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(m-1) \end{bmatrix}, \begin{bmatrix} u(m) \\ u(m+1) \\ \vdots \\ u(2m-1) \end{bmatrix}, \ldots\ldots \right\} \in \mathbf{V}^m \qquad \textbf{(2-4)}$$

referring to the period mh is known as the lifted signal. The lifting operator L can be considered as a transformation matrix. It has the property of neither being lower triangular nor *Toeplitz*. The system L is thus non-causal and time-varying. It can be shown that L is norm preserving; due to the property of *isometry*. The inverse or *adjoint* operator of lifting (L^{-1}) exits as $LL^{-1} = L^{-1}L = I$. The Figure 2-4 gives the block diagram description of both lifting and its inverse. Slow-rate signals are represented as low-frequency dots; the fast-rate signals as high-frequency dots.

Lifting Discrete-time Systems

A periodic system $\mathbf{P}$ can similarly be lifted to a shift invariant system $\underline{\mathbf{P}}$ of higher dimension as $\underline{\mathbf{P}} = L^{-1}\mathbf{P}L$. The shaded region in Figure 2-4 represented the lifted system $\underline{\mathbf{P}}$ with lifted input and output signals.

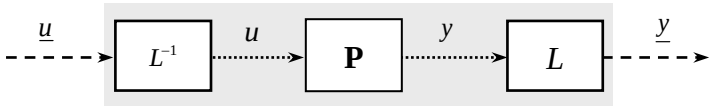


Figure 2-4: Lifting discrete-time system

Theorem: Given the transfer function of $\mathbf{P}$ in terms of state-space data as

$$\mathbf{p}(z) = \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right]$$

the *m* dimensional lifted system $\underline{\mathbf{P}}$ is FDLTI, with transfer function given as

$$\underline{\mathbf{p}}(z) = \left[\begin{array}{c|cccc} \mathbf{A}^m & \mathbf{A}^{m-1}\mathbf{B} & \mathbf{A}^{m-2}\mathbf{B} & \ldots & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} & 0 & \ldots & 0 \\ \mathbf{CA} & \mathbf{CB} & \mathbf{D} & \ldots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ \mathbf{CA}^{m-1} & \mathbf{CA}^{m-2}\mathbf{B} & \mathbf{CA}^{m-3}\mathbf{B} & \ldots & \mathbf{D} \end{array}\right]$$

□

Since lifting operation preserves norms, it follows that the norms of the two transfer functions, $\mathbf{p}(z)$ and $\underline{\mathbf{p}}(z)$, hold following two relationship

1. $\|\mathbf{p}\|_2^2 = \dfrac{\|\underline{\mathbf{p}}\|_2^2}{m}$
2. $\|\mathbf{p}\|_\infty = \|\underline{\mathbf{p}}\|_\infty$

The lifted representation is somewhat similar to the 'expanded' representation used in [22] but lifting requires no increase in state dimensions of the system.

Plant Model in Lifted Domain

An important and useful application of lifting is to extract single rate models out of multirate systems making them amenable to simple analysis and design techniques. Consider a simple multirate discrete-time plant model expressed by state-space model as

$$\begin{aligned} \mathbf{x}[i, j+1] &= \mathbf{\Phi}\,\mathbf{x}[i, j] + \mathbf{\Gamma}\,u[i, j] \\ y[i,0] &= \mathbf{H}\,\mathbf{x}[i,0] \end{aligned} \qquad \textbf{(2-5)}$$

where $j = 0,1,2,.....,m-1$

The model clearly present the multirate nature of the system where state update takes place at every j^{th} control instant, while output update occurs after as many control inputs as the input multiplicity m. The lifted multirate plant model is obtained by means of progressive backward substitutions as follows:

$$\begin{aligned} \mathbf{x}[i+1,0] &= \mathbf{\Phi}\mathbf{x}[i,m-1] + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \\ &= \mathbf{\Phi}\,(\mathbf{\Phi}\,\mathbf{x}[i,m-2] + \mathbf{\Gamma}\,\mathbf{u}[i,m-2]) + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \\ &= \mathbf{\Phi}^2\mathbf{x}[i,m-2] + \mathbf{\Phi}\mathbf{\Gamma}\,\mathbf{u}[i,m-2] + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \\ &= \mathbf{\Phi}^2\,(\mathbf{\Phi}\,\mathbf{x}[i,m-3] + \mathbf{\Phi}\mathbf{\Gamma}\,\mathbf{u}[i,m-3]) + \mathbf{\Phi}\mathbf{\Gamma}\,\mathbf{u}[i,m-2] + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \\ &= \mathbf{\Phi}^3\mathbf{x}[i,m-3] + \mathbf{\Phi}^2\mathbf{\Gamma}\,\mathbf{u}[i,m-3] + \mathbf{\Phi}\mathbf{\Gamma}\,\mathbf{u}[i,m-2] + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \\ &= ..... \\ &= \mathbf{\Phi}^m\mathbf{x}[i,0] + \mathbf{\Phi}^{m-1}\mathbf{\Gamma}\,\mathbf{u}[i,0] + ... + \mathbf{\Phi}\mathbf{\Gamma}\,\mathbf{u}[i,m-2] + \mathbf{\Gamma}\,\mathbf{u}[i,m-1] \end{aligned}$$

The equivalent single-rate model in lifted domain can thus be written as

$$\begin{aligned} \mathbf{x}[i+1,0] &= \underline{\mathbf{\Phi}}\,\mathbf{x}[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \\ y[i,0] &= \underline{\mathbf{H}}\,\mathbf{x}[i,0] \end{aligned} \qquad \textbf{(2-6)}$$

where $$\left[\begin{array}{c|c} \underline{\underline{\mathbf{\Phi}}} & \underline{\mathbf{\Gamma}} \\ \hline \underline{\mathbf{H}} & \mathbf{O} \end{array}\right] = \left[\begin{array}{c|ccc} \mathbf{\Phi}^{m} & \mathbf{\Phi}^{m-1}\mathbf{\Gamma} & \mathbf{\Phi}^{m-2}\mathbf{\Gamma} & \cdots \quad \mathbf{\Gamma} \\ \hline \mathbf{H} & \multicolumn{3}{c}{\mathbf{O}} \end{array}\right]$$

and $$\underline{\mathbf{u}}[i,0] = \begin{bmatrix} \mathbf{u}[i,0] \\ \mathbf{u}[i,1] \\ \vdots \\ \mathbf{u}[i,m\text{-}1] \end{bmatrix} \in \Re^{m \times n_y} \qquad \textbf{(2-7)}$$

The triplet $(\underline{\underline{\mathbf{\Phi}}}, \underline{\mathbf{\Gamma}}, \underline{\mathbf{H}})$ can be shown to be controllable and observable, given $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{H})$ is controllable and observable (see Appendix A for proof).

Chapter 3

Visual Servoing

The idea of a machine, able to sense and interact with the external environment having some degree of autonomy, is realized in the form of robots. Sensors and devices, which could bestow such a machine with external information, became one of the major areas of interest for the enthusiasts. Sensor design, performance, type, and like issues were readily explored. Camera, as a vision sensor, fascinated researchers for two main reasons [8]:

- First, there seemed tremendous potential to extract wide range of information from a set of images. Vision sensors were thus much more cost effective.
- Second, a robot having visual capability brings in near human-like behavior strengthening the concept of humanoid/behavioral robots. Thus considerably increasing the versatility and application domain of robots.

Vision is a non-contact sensing modality. It enables a robot to acquire dense information about the environment in the form of captured scenes. Vision geometry, image acquisition, analysis, feature detection, feature matching etc. serve a lot to *understand* the external environment and then *decide* how to *act*. Naturally the field of vision-based control proved to be a merger of several elementary areas including image processing and computer vision, kinematics, dynamics, control theory and real time computing, today commonly referred as *visual servoing.*

Until now, the hardware costs for a real-time visual servo system were very high. However, with technological advancement the visual processing and control calculations required for visual servoing can now be performed on PCs. It is now possible to construct modular visual servoing systems in software that operate at or near camera frame rate.

The combination of robot control and computer vision can become extremely important, from applications point of view, when dealing with a robotic systems working in uncertain and dynamic environments. Many practical applications of visual servoing have been demonstrated, some of these include: picking and placing parts from a conveyer belt, picking fruits, pan-tilt tracking by an active-head, track following by a mobile robot, arc welding, seam and arc welding of steel sheets etc. [8].

3.1 Brief Literature Survey

The vision-based control has more than 30 years of research history. One of the earliest significant work was by Shirai and Inoue published in 1973, inspiring considerable number of efforts towards the visual control of robot manipulators [8]. They described how a visual feedback loop can be used to correct the position of a robot to increase task accuracy. With limited visual processing capabilities, vision routines and the robot control routines were performed in a sequential manner and robot must come to a complete stop before the camera could acquire a new image. Due to technological limitations, this and some other earlier approaches are classified as *static look-then-move* approach [9]. The *dynamic* look-and-move feature-based visual servo architecture, a popular solution proposed by Weiss et al. [23], performs vision and robot control routines in parallel. It typically incorporates an outer vision loop and inner joint sensor feedback loop. J. Hill and W. T. Park in their renowned work described the real-time control of a Unimate robot with a mobile camera. The term ‘visual servoing’ is also attributed to Hill and Park [8].

It is easy to figure out that a vast majority of literature on visual servoing shows concern about visual servoing *kinematics,* investigated in large part by computer vision researcher. There has been little emphasis on *dynamic* performance of a visual servo system throughout, which resulted in poor control performance of visual dynamic systems [8], [24]. During the past decade tracking performance, robustness and stability of visual servo systems emerged as an active issue. The research community started considerable efforts since early 90’s in improving visual servo dynamics concentrating on the control structure of such systems. The work of Peter Corke [25], [26] is much significant in this regard. He pointed out the limitations of feedback control and emphasized on feedforward control. Chroust et al. [27] also proposed higher sampling rate and feedforward control for smooth tracking.

Design of controllers has usually been done by conventional PID control. However other readily utilized control strategies are LQG and Adaptive control. Papanikolopoulos et al. utilized self-tuning regulator (an adaptive control strategy) in [28] and a discrete LQG controller in [29] for computing the desired trajectory of a camera mounted manipulator. They use sum of squared difference (SSD) optical flow for computation of vector of discrete displacement so that system states and exogenous disturbance can be estimated. Recently, the successful applications of Model Predictive control (MPC) in process industry urged many to benefit robot motion control from those experiences. Barreto el. al. [30], [31] successfully implemented active visual tracker through dynamic matrix control (an MPC algorithm). Kalman filter approach has been widely used to construct the system states and for disturbance estimation [8], [24], [28][31]. However, Allen et al. [32] showed that the white-Gaussian noise assumptioin by Kalman filter cannot be realistic while tracking and grasping moving objects. They utilized a fixed gain $\alpha - \beta - \gamma$ filter for prediction and a *moving-average* filter for smoothing.

Above mentioned implementations gave reasonable results to cope with latency and other dynamic issues. However, the multirate-control approach for visual servo systems is not as thoroughly investigated as its need was mentioned, e.g. [8]. Fujimoto et al. [10] successfully implemented multirate digital control for visual tracking by an end-effector mounted planar robot.

3.2 Vision Sensor and its Configurations

The earliest reports on vision-based control used *vidicon* image sensors. These devices had a number of undesirable features like weight, bulk, fragility, poor image stability etc. CCD or CID sensor based cameras became popular since mid 1980's. Area sensors are most common, however a few reports of line sensors are present [8].

When tracking moving objects motion blur can be a significant problem. The object can appear elongated in the direction of travel, and the centroid of the blurred object thus lags the real object's centroid. A simple binary vision system with fixed threshold may lose sight of a rapidly moving target. However, most modern CCD cameras available today have an electronic shuttering facility and use a short exposure time, which can reduce the blur effect.

Cameras can generally be either fixed or mounted on the robot's end-effector, the later also known as *eye-in-hand* configuration. The first configuration has the cameras fixed in the workspace. In this case the camera is related to the base coordinate system (fixed base) of the robot. The image of the target is thus independent of the robot motion, unless the target is the end-effector itself. Nearly all stereo-vision based approaches appearing proposed in literature use fixed camera configuration. For eye-in-hand configuration, there exists a known relationship between the pose of the camera and the end-effector. The benefits of this configuration include the ability to avoid occlusion, resolve ambiguity and increase accuracy by directing its attention. Since the target position is not directly measurable, an eye-in-hand camera, however, can provide the difference between the end-effector and target position [8], [27].

Another but less common configuration is for the camera to be mounted on another robot or pan-tilt head in order to observe the visually controlled robot for some sort of supervisory control. There are reports of hybrid approaches, using both eye-in-hand and fixed cameras. In all configurations, camera calibration is necessary.

3.3 Visual Servoing Architectures

Visual servoing architectures can be categorized according to following accepted rule: If the control structure is hierarchical with outer visual servo loop passing commands to the inner level joint position loop, then such control architecture is called *dynamic look and move* architecture. The vision system provides set-point inputs to the joint-level controller thus making use of joint feedback to internally stabilize the robot. On the other hand, if there is no control hierarchy then it is called *direct visual servo* architecture. Direct visual servo eliminates the joint controller entirely replacing it with a visual servo controller that uses vision alone to stabilize the mechanism.

In this thesis we consider dynamic look and move architecture. Dynamic look-and-move approach greatly simplifies the problem, and allows simple construction. Nearly all implemented systems adopt the dynamic look-and-move approach. Relatively low sampling rates available from vision, makes direct visual servo control of a robot end-effector, with complex nonlinear dynamics, an extremely challenging control problem.

Both of these architectures can be further classified into position and image based control architectures, depending upon the error-space of the signal fed to the visual servo

controller. Rest of this section provides a brief account of these two visual servo architectures.

3.3.1 Position-based Visual Servoing

Position-Based Visual Servo (PBVS), in which computer vision techniques are used to reconstruct a representation of the 3D workspace of the robot, and actuator commands are computed with respect to the 3D workspace. Features are extracted from the image and used to estimate the pose of the target with respect to the camera. Using these values, an error between the current and the desired pose of the robot is defined in the task space.

Position-based visual servo architectures as described in [8], [23] are shown below

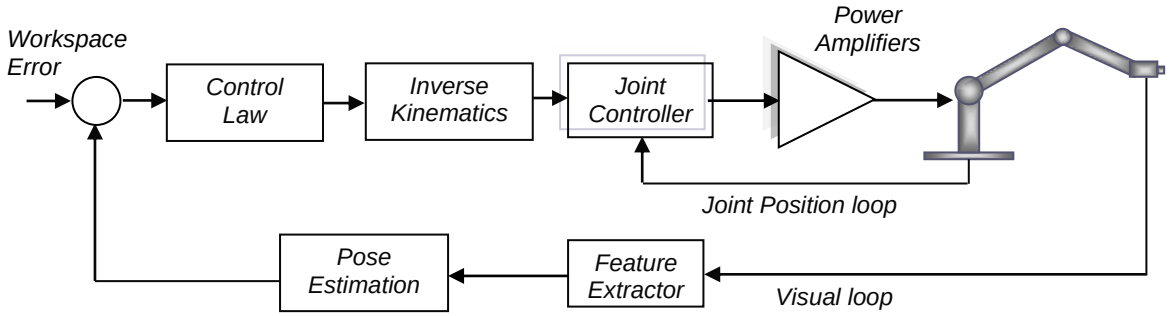


Figure 3-1: Dynamic position-based look-and-move structure

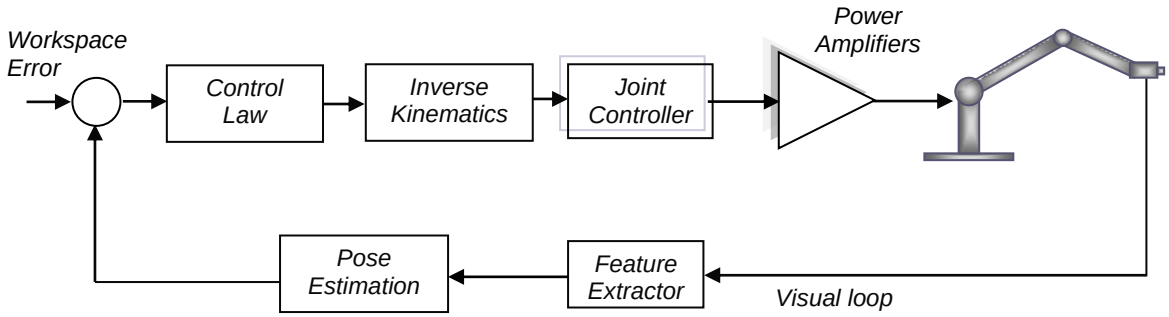


Figure 3-2: Position-based direct visual servo structure

The principle advantage of PBVS is that it is possible to describe tasks in terms of positioning in Cartesian coordinates. However it is often highly calibration dependent. In an environment where moderate positioning accuracy is required, extant calibration techniques may provide a sufficiently accurate solution. However, if the cameras are

moving and high accuracy is required, calibration sensitivity is an important issue. Endpoint closed- loop systems are, however, less sensitive to calibration.

3.3.2 Image-based Visual Servoing

In image-based visual servo control the error signal is defined directly in terms of image feature parameters in contrast to position-based methods that define the error signal in the task space coordinates. The system may use either a fixed camera or an eye-in-hand configuration. In either case motion of the manipulator causes changes to the image observed by the vision system. Thus an appropriate error function has to be determined such that it returns zero when the task is achieved.

Image-based visual servo architectures as described in [8], [23] are shown below

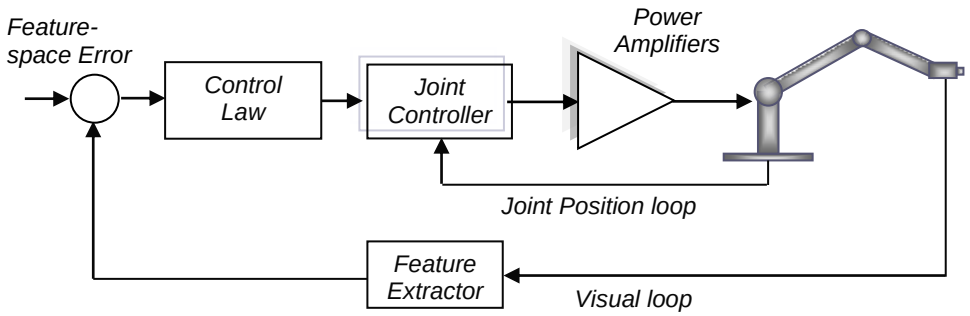


Figure 3-3: Dynamic image-based look-and-move structure

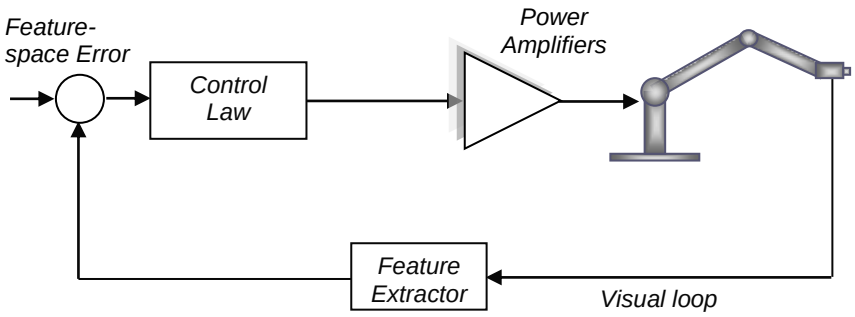


Figure 3-4: Image-based direct visual servo structure

Although the error is defined on the image parameter space, the manipulator control input is typically defined either in joint coordinates or in task space coordinates. Therefore it is necessary to relate differential changes in the image feature parameters to differential

changes in the position of the manipulator. The *image Jacobian* captures these relationships. Image Jacobian was first introduced by Weiss et al. [23]. Details of the image-Jacobian can be found in [8].

3.4 Visual Servoing as Control Problem

Visual servo systems use vision feedback to generate set-points which are sent to the joint controllers (in a dynamic look-then-move fashion). While tracking an object, its image should typically lie at the center of the image plane, the process called fixation. Any displacement from the image center can be modeled as disturbance (see Figure 3-5).

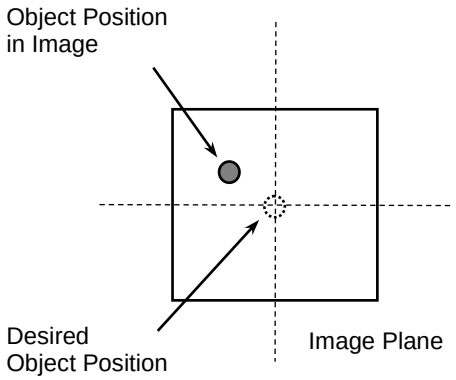


Figure 3-5: Object displacement from desired position acts as disturbance

The response of the object image centroid to its reference position (center of image) can be given by

$$\frac{Y(z)}{O(z)} = \frac{H(z)G(z)F(z)}{1+H(z)G(z)F(z)}$$

where $Y(z)$ is the actual object position in image and $O(z)$ is the desired object position in image. $F(z)$, $G(z)$ and $H(z)$ represent the transfer function of the controller, robot and camera respectively. The control problem thus transforms into a regulation problem.

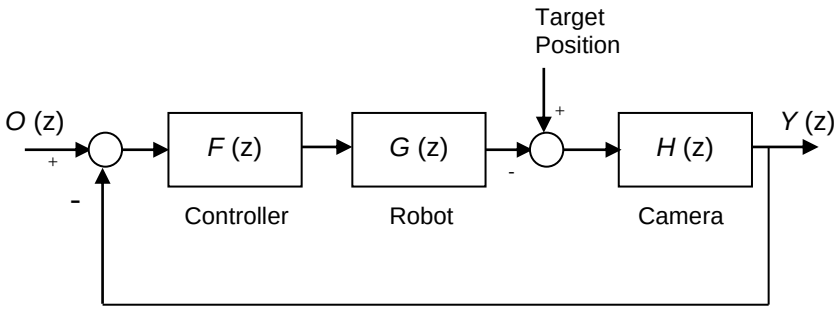


Figure 3-6: Visual servoing control structure

Some of the classical approaches in order to achieve improved tracking performance are [8], [24]:

- Increasing the loop gain, which minimizes the magnitude of ramp following error. However this gain is severely constrained for systems having open-loop delay.
- Increasing the *Type* of the system. This is achieved by adding open-loop integrators. Type 1 and Type 2 systems have zero steady state error to a step and ramp demand respectively.
- Introducing feedforward of the signal to be tracked. For the visual servo systems, the signal to be tracked is the target position which is not directly measurable. This approach requires filtering and estimation techniques.

Chapter 4

Multirate Control of Visual Servo Systems

4.1 Multirate Nature of Visual Servo Systems

As discussed earlier, there exist hardware and system dynamic constraints that dictate the sampling rate selection for a particular signal. The control input period is generally decided by the speed of the actuator, D/A converter, or the computational time complexity, whereas the output sampling period could be constrained by the speed of the sensor or the A/D converter. Many practical systems usually hold such restrictions on input/output sampling rates.

Visual servo systems incorporate vision sensor as output measurement device. Many industrial vision systems utilize CCD video cameras as vision sensor, which is inherently a digital sensor with limited output rate. Commonly available cameras generally conform to either of the two main video standards: RS170 (USA, Canada, Japan) or CCIR (Europe, Australia) with frame rates of 30 Hz and 25 Hz respectively [8]. Most vision based control systems are used in conjunction with a robot arm or a pan-tilt head, where the encoder at the joint servo level can normally update the position information at 1000 – 5000 Hz [10], [35]. The joint control loop can thus operate at much higher rate than the outer visual servo loop. The inclusion of two feedback loops operating at different rates makes a vision-based control problem as an inherent multirate system problem.

Also because image processing algorithms are always time consuming, the overall visual cycle is much longer relative to a desired control interval. If the control interval is

prolonged to match the availability of measurements, the control performance deteriorates significantly [4], [10]. On the other hand, if system is fast sampled, it may make the discrete time model non-minimum phase when the original continuous time system has a relative degree higher than two [5]. Although single-rate solutions incorporating filtering and estimation of system states are available, but multirate control approach seems to be a natural solution.

4.2 Prior Work

Investigation of the capabilities of multirate control theory towards visual servo control is an open research area [8]. Quite a few efforts are reported in this connection.

Nemani et al. [36] proposed one of the first multirate formulations, using lifting technique, based on modern sampled-data control theory. They formulated an *l*-1 norm optimal control problem to minimize the maximum time-domain error, which has direct connection to camera FOV and mechanical latency. They presented and numerical example to demonstrate the design advantage ir multirate formulation was based on discrete-time lifting technique.

Niwa et al. [34] developed a sensor fusion algorithm for visual control of mobile robots with a multirate version of Kalman filter having time-varying filter gain matrix.

Fujimoto et al. [10] proposed a multirate scheme with focus on perfect disturbance rejection at intersample points. Extension to systems with time delay was made based on observer included time-delay model. The proposed multirate scheme was applied for 2D planar visual servoing. A recent work by Fujimoto [35] addresses the more complex problem of visually servoing a 6-DOF manipulator.

4.3 Modeling of Visual Servo System

The vision sensor is generally not associated with any dynamic and it essentially converts the visual information into electrical signals. Associated with the camera is the vision system which computes the relative pose (position and orientation) of the target with respect to the camera after processing the image. If the mapping from the object space to the image plane were unique and independent of the orientation of the object, the transformation from the input to the output of the camera would be the identity matrix. However this might not be the case as the mapping is position dependent and can be non

unique with respect to the depth of the object. Also there is a significant delay associated with the image processing unit, as image processing is computationally intensive.

An important task in visual servoing is the tracking of a moving target. While tracking an object, its image should typically lie at the center of the image plane, a process called *fixation*. In order to ensure proper fixation, the target features are to be clearly identified in the image to calculate the error from the desired pose. Assume that *k* feature points on the target are selected, and defined as

$$\boldsymbol{\eta} := \begin{bmatrix} \boldsymbol{\eta}_1^T & \boldsymbol{\eta}_2^T & \dots & \boldsymbol{\eta}_k^T \end{bmatrix}^T, \quad \text{where} \quad \boldsymbol{\eta}_i = \begin{bmatrix} \eta_{xi} \\ \eta_{yi} \end{bmatrix}, \quad i = 1, 2, ..., k$$

Let R denote the reference coordinate system, usually the workspace. The position and orientation (pose) of any point $\mathbf{p}$ in R is represented by a 6-elemental vector. Let ${}^R\mathbf{p}_c \in \Re^6$ and ${}^R\mathbf{p}_o \in \Re^6$ be the pose of the camera and the object respectively. The camera is modeled by a mapping $\sigma : \Re^6 \times \Re^6 \to \Re^{2k}$ as

$$\boldsymbol{\eta} = \sigma({}^R\mathbf{p}_c, {}^R\mathbf{p}_o) \qquad \textbf{(4-1)}$$

The derivative of (4-1) is given by

$$\dot{\boldsymbol{\eta}} = \frac{\partial \sigma}{\partial\, {}^R\mathbf{p}_c} {}^R\dot{\mathbf{p}}_c + \frac{\partial \sigma}{\partial\, {}^R\mathbf{p}_o} {}^R\dot{\mathbf{p}}_o = \mathbf{J}(\boldsymbol{\eta}, \mathbf{z})\, \boldsymbol{\upsilon}_c + d(t) \qquad \textbf{(4-2)}$$

where $\mathbf{J} = [\mathbf{J}_1^T, ..., \mathbf{J}_k^T]^T$ is the image-Jacobian [8], $\mathbf{z} = [z_1, ..., z_k]^T$ is the distance between the k^{th} feature point and the camera, $\boldsymbol{\upsilon}_c = {}^R\dot{\mathbf{p}}_c$ is the velocity screw of the camera frame expressed in camera coordinates, and $d(t)$ is the disturbance caused by the object motion. The image-Jacobian relates differential changes in the image feature parameters to differential changes in the position of the manipulator, as given by following relation

$$\dot{\boldsymbol{\eta}} = \mathbf{J}_k\, {}^R\dot{\mathbf{p}}_e \qquad \textbf{(4-3)}$$

Here ${}^R\dot{\mathbf{p}}_e$ denote the end-effector velocity. Above relationship allows determining the manipulator velocity required to achieve some desired value of the image feature velocity.

4.4 The Multirate Control Design

A multirate control design is formulated in this section. To our knowledge, this design formulation is not used for control design of visual servo systems. We assume that the plant of concern is LTI, and SISO system. Generalization for MIMO systems can be easily obtained.

Consider an n^{th} order continuous-time plant described by the following state-space representation

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ y &= \mathbf{C}\mathbf{x} \end{aligned} \qquad \textbf{(4-4)}$$

where $\mathbf{x} \in \Re^n$ is the system state vector, $u \in \Re$ is the control input, $y \in \Re$ is the output vector. The matrix dimensions are, $\mathbf{A} \in \Re^{n\times n}$, $\mathbf{B} \in \Re^{n\times 1}$ and $\mathbf{C} \in \Re^{1\times n}$. It is assumed that the pair $(\mathbf{A},\mathbf{B})$ is *controllable*, i.e. the *controllability matrix* $\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix}$ is of full rank. Also the pair $(\mathbf{A},\mathbf{C})$ is *observable*, which implies that the *observability matrix* $\begin{bmatrix} \mathbf{C} & \mathbf{CA} & \mathbf{CA}^2 & \dots & \mathbf{CA}^{n-1} \end{bmatrix}^T$ is invertible.

4.4.1 Controller Design

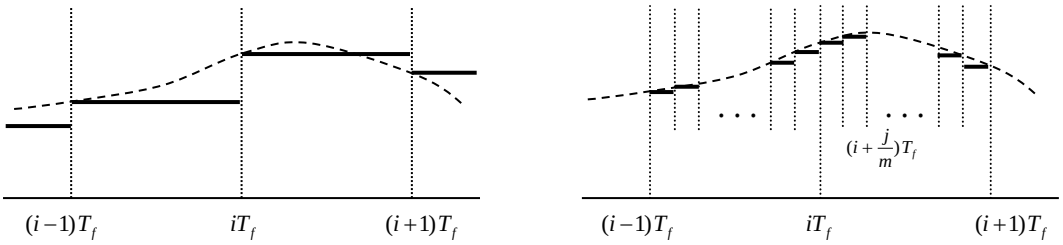
The multirate state-feedback controller design is presented after [11]. Let the plant output is sampled at a slower rate T_f while controller sends the control commands to the plant at higher control rate T_c. It is assumed, for convenience, that input multiplicity $T_f / T_c = m$ is an integer with $m > 1$. The control input to the plant in i^{th} frame period can be represented as a lifted signal

$$\underline{\mathbf{u}}[i,0] = \left[u[i,0], u[i,1], \ldots\ldots\ldots, u[i,m-1]\right]^T$$

where $u[i,j] = u\left((i+\mu_j)T_f\right) \quad \mu_j = \dfrac{j}{m}, \qquad j = 0,1,2,\ldots\ldots m-1$

According to (2-7) the control input vector can be represented in lifted form as

$$\underline{\mathbf{u}}[i,0] = L\,u[i,j] \qquad \textbf{(4-5)}$$



(a) **(b)**

Figure 4-1: Description of hold operations (a) Zero-Order Hold (b) Multirate Hold

The simple multirate discrete-time plant model can be expressed as

$$\begin{aligned} \mathbf{x}[i, j+1] &= \mathbf{\Phi}\,\mathbf{x}[i, j] + \mathbf{\Gamma}\,u[i, j] \\ y[i,0] &= \mathbf{H}\,\mathbf{x}[i,0] \end{aligned} \qquad \textbf{(4-6)}$$

where $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{H})$ provide discrete-time state space representation of the plant model based on input sampling rate T_c, as given by

$$\mathbf{\Phi} = e^{\mathbf{A}T_c}$$

$$\mathbf{\Gamma} = \int_0^{T_c} e^{\mathbf{A}(T_c-\eta)}\mathbf{B}\,d\eta$$

$$\mathbf{H} = \mathbf{C}$$

Equations (4-6) clearly present the multirate nature of the model. Note that the state update takes place at every j^{th} control instant, so is the control input (see Figure 4-1). However output update occurs after *m* control inputs to the plant. Assuming that all plant states are available at our disposal, the state-feedback control law is given by

$$u[i, j] = -\mathbf{K}\,\mathbf{x}[i, j] \qquad \textbf{(4-7)}$$

therefore $\mathbf{x}[i, j+1] = (\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K})\,\mathbf{x}[i, j]$

and by mean of progressive substitution, clearly

$$\mathbf{x}[i+1,0] = (\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K})^m\,\mathbf{x}[i,0] \qquad \textbf{(4-8)}$$

The lifted multirate plant model obtained from Equation 4-6 by means of progressive substitution is given by

$$\begin{aligned} \mathbf{x}[i+1,0] &= \underline{\mathbf{\Phi}}\,\mathbf{x}[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \\ y[i,0] &= \underline{\mathbf{H}}\,\mathbf{x}[i,0] \end{aligned} \qquad \textbf{(4-9)}$$

where $$\left[\begin{array}{c|c} \underline{\mathbf{\Phi}} & \underline{\mathbf{\Gamma}} \\ \hline \underline{\mathbf{H}} & \mathbf{O} \end{array}\right] \Rightarrow \left[\begin{array}{c|ccc} \mathbf{\Phi}^{m} & \mathbf{\Phi}^{m-1}\mathbf{\Gamma} & \mathbf{\Phi}^{m-2}\mathbf{\Gamma} & \cdots \quad \mathbf{\Gamma} \\ \hline \mathbf{H} & \multicolumn{3}{c}{\mathbf{O}} \end{array}\right] \qquad \textbf{(4-10)}$$

The triplet $(\underline{\mathbf{\Phi}}, \underline{\mathbf{\Gamma}}, \underline{\mathbf{H}})$ can be shown to be controllable and observable, given $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{H})$ is controllable and observable (see Appendix A for proof). The state-feedback control law is given by

$$\underline{\mathbf{u}}[i,0] = -\underline{\mathbf{K}}\,\mathbf{x}[i,0] \qquad \textbf{(4-11)}$$

$$\underline{\mathbf{K}} = \left[K_0,\ K_1,\ K_2, \ldots\ldots, K_{m-1}\right]^T \in \Re^{n_u m \times n}$$

Substituting (4-11) in (4-9)

$$\mathbf{x}[i+1,0] = (\underline{\mathbf{\Phi}} - \underline{\mathbf{\Gamma}}\,\underline{\mathbf{K}})\,\mathbf{x}[i,0] \qquad \textbf{(4-12)}$$

Equating (4-8) and (4-12), the state-feedback gain $\underline{\mathbf{K}}$ is computed as

$$\underline{\mathbf{\Phi}} - \underline{\mathbf{\Gamma}}\,\underline{\mathbf{K}} = (\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K})^{m}$$

$$\Rightarrow \qquad \underline{\mathbf{K}} = \underline{\mathbf{\Gamma}}^{+}\left(\underline{\mathbf{\Phi}} - (\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K})^{m}\right) \qquad \textbf{(4-13)}$$

where $\underline{\mathbf{\Gamma}}^{+}$ denotes the *pseudo-inverse* of $\underline{\mathbf{\Gamma}}$. Given $\mathbf{K}$ designed for $(\mathbf{\Phi}, \mathbf{\Gamma})$, above multirate state-feedback scheme delivers closed loop discrete state-matching.

4.4.2 Observer Design

In order to implement above state feedback controller, the system state have to be reconstructed. A full state observer is designed for this purpose. In a simple state observer, the plant output serves as input to the observer for state estimation. State equation for a simple SISO prediction observer can be written as [3]

$$\hat{\mathbf{x}}[i] = \left(\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K} - \mathbf{L}_p\mathbf{H}\right)\hat{\mathbf{x}}[i-1] + \mathbf{L}_p\, y[i-1]$$

$$u[i] = -\mathbf{K}\,\hat{\mathbf{x}}[i]$$

However, an observer design constraint, in case of visual servo system, is that direct plant output is not available. The vision sensor (camera) measures only the error between the current object and camera positions. Hence the observer has to recover the system state from the output error command (see Figure 4-2).

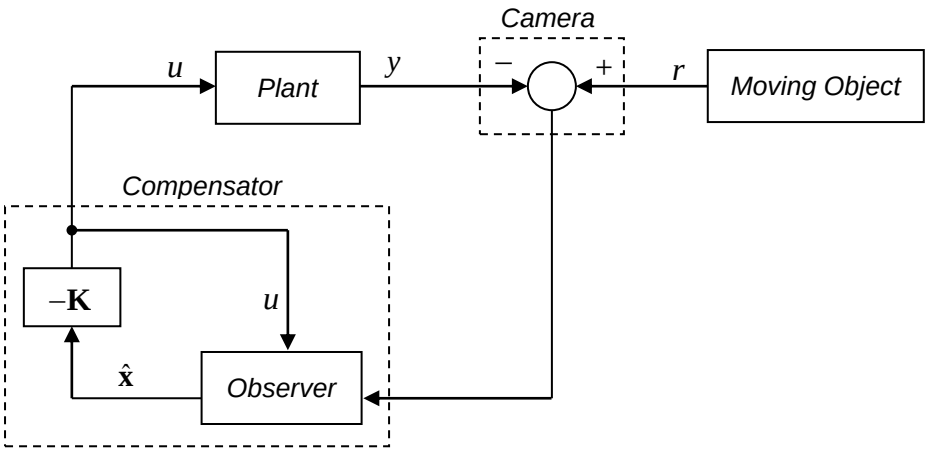


Figure 4-2: Output error as input to the observer

However, the observer with output-error input provides faster response but with relatively higher control usage than simple state command structure [2], [3]. To reduce the effect of estimation errors, we select *current* state observer structure. For design purpose, we consider simple state command structure. The formulation can be easily extended by replacing $y[i]$ with $r[i]-y[i]$.

Consider a conventional multirate current state observer for (4-6), described by

$$\begin{aligned}\hat{\mathbf{x}}[i, j+1] &= \mathbf{\Phi}\,\bar{\mathbf{x}}[i, j] + \mathbf{\Gamma}\,u[i, j] \\ \bar{\mathbf{x}}[i, j] &= \hat{\mathbf{x}}[i, j] + \mathbf{L}\,(y[i,0] - \mathbf{H}\,\hat{\mathbf{x}}[i,0])\end{aligned} \qquad \textbf{(4-14)}$$

where $\mathbf{L}$ is the current stare observer gain. Given the lifted system model (4-9), the current state observer is given by

$$\begin{aligned}\hat{\mathbf{x}}[i+1,0] &= \underline{\mathbf{\Phi}}\,\bar{\mathbf{x}}[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \\ \bar{\mathbf{x}}[i,0] &= \hat{\mathbf{x}}[i,0] + \underline{\mathbf{L}}\,(y[i,0] - \underline{\mathbf{H}}\,\hat{\mathbf{x}}[i,0])\end{aligned} \qquad \textbf{(4-15)}$$

$$\Rightarrow \qquad \hat{\mathbf{x}}[i+1,0] = (\underline{\mathbf{\Phi}} - \underline{\mathbf{L}}\,\underline{\mathbf{H}})\,\hat{\mathbf{x}}[i,0] + \underline{\mathbf{L}}\,y[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \qquad \textbf{(4-16)}$$

The current state observer gain $\underline{\mathbf{L}}$ is determined of discrete state matching at each control instant. Let $e[i,0] = y[i,0] - \mathbf{H}\hat{\mathbf{x}}[i,0]$ in (4-14). Progressive substitution results as follows

$$\begin{aligned}\hat{\mathbf{x}}[i+1,0] &= \mathbf{\Phi}\,\hat{\mathbf{x}}[i,m-1] + \mathbf{\Phi}\,\mathbf{L}\,e[i,0] + \mathbf{\Gamma}\,u[i,m-1] \\ &= \mathbf{\Phi}^2\,\hat{\mathbf{x}}[i,m-1] + (\mathbf{\Phi}^2 + \mathbf{\Phi})\,\mathbf{L}\,e[i,0] + \begin{bmatrix}\mathbf{\Phi}\mathbf{\Gamma} & \mathbf{\Gamma}\end{bmatrix}\begin{bmatrix}u[i,m-2] \\ u[i,m-1]\end{bmatrix} \\ &= \text{.....}\end{aligned}$$

which finally reduces to

$$\hat{\mathbf{x}}[i+1,0] = \left(\underline{\mathbf{\Phi}} - \left(\sum_{k=1}^{m}\mathbf{\Phi}^k\right)\mathbf{L}\,\mathbf{H}\right)\hat{\mathbf{x}}[i,0] + \left(\sum_{k=1}^{m}\mathbf{\Phi}^k\right)\mathbf{L}\,y[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \quad \textbf{(4-17)}$$

Comparing (4-16) and (4-17), it is obvious that

$$\underline{\mathbf{L}} = \underline{\mathbf{\Phi}}^{-1}\left(\sum_{k=1}^{m}\mathbf{\Phi}^k\right)\mathbf{L} \quad \textbf{(4-18)}$$

(4-18) produces discrete state matching of (4-15) with (4-14) at each sampling update instant.

Chapter 5

Simulation Setup

The objective of the simulation work is to compare the response of conventional single-rate control with that of multirate-based visual tracking system. In order to focus on the control aspect, the visual servoing problem is restricted to 2-D visual tracking.

The simulation environment is MATLAB® and SIMULINK® (all from The MathWorks Inc.). The Control System Toolbox® is utilized for control calculations. The whole visual tracking system is modeled in SIMULINK. The Robotics Toolbox® provided simple way to simulate the kinematic and dynamic behavior of robotic manipulator [37] (see Appendix C for details).

5.1 Moving Target

The target is considered to be moving in a plane parallel to the camera plane (see Figure 5-2). The movement path can be arbitrary, however it is assumed that there is no sharp turn. Tracking moving object with sharp corners in its path may cause loss of sight by vision sensor, system instability, poor tracking etc. and is still an open problem. The different target paths considered for simulation purpose include straight line, circular and sinusoidal paths. Some other polynomial fitted paths have also been tested. We also assume that the speed of the moving object does not exceed a predefined limit. The preset target upper speed limit while simulation is 2 m/s. The shape of the object is uniform such that its centroid calculation from image is trivial. The object is allowed to move just below the camera at a specified depth in the shaded area shown in so as to avoid singularities and loss of view.

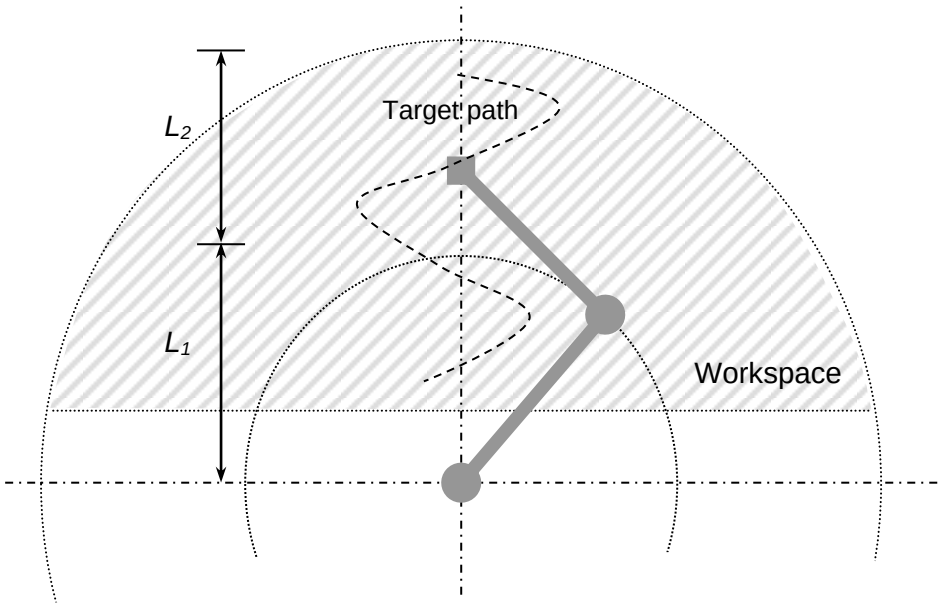


Figure 5-1: Target moves in the shaded region to avoid loss of view

5.2 Robotic System

The robot instance simulated is a fixed-base direct-drive two link planar robot (RR-configuration) [10], [38]. Simplicity in robot architecture allows focusing on dynamical aspects of multirate systems. The camera is mounted on the robot end-effector (eye-in-hand configuration) actively tracking a moving target in a parallel plane as shown in Figure 5-1. Here q_1 and q_2 represent the joint angles with respect to the reference base frame. We have used D – H representations. The kinematic and dynamic models of two-link planar robot are given in Appendix B (further details can be found in [39]). The manipulator parameters are given in Appendix D.

The motor armature shaft and position/speed sensor shaft inertia is ignored since it is usually a much smaller quantity compared to link inertia. It is assumed that no dynamics are associated with joint sensors (encoders) and sensor gains are thus considered unity. Further, it is assumed that the target only moves in the robot workspace in a preplanned area (Figure 5-1) so that the non-singularity of manipulator *Jacobian* is assured. In presence of singularities, it is not possible to give any arbitrary motion to the end-effector,

since they represent the configurations where structural mobility of the manipulator is restricted.

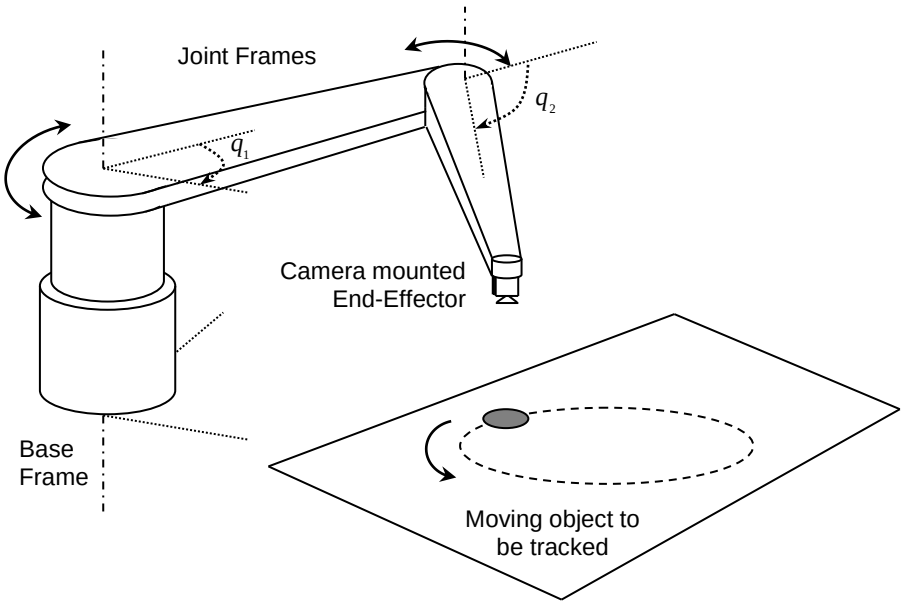


Figure 5-2: Visual tracking using a 2-link planar robot

5.3 Vision System

It is assumed that a CCD camera is mounted on the robot end-effector. The image plane resolution is 512×512 pixels. Pixels are normally rectangular, instead of square. However, for convenience we assume that a pixel is a square. The pixel scale factor of 80/mm × 80/mm is considered. The focal length *f* is taken as 8mm, and the target depth *z* is taken as 500mm. These specifications differ slightly from the data provided by [26], [37]. The camera field of view (FOV) is 400mm × 400mm.

For perfect tracking, it is required that origin of camera coordinate system should lie exactly above the object centroid (the selected feature). The desired feature point position in the image is the center of the image plane. Any drift from the center of image can be modeled as disturbance in the system. Figure 5-3 shows the orientation of camera, image and target plane in the simulation setup. The optical axis of the camera is perpendicular to camera and target plane. The distance *z* between these two planes is constant and known.

We assume pinhole camera model with perspective projection. Let $\boldsymbol{\eta}$ denote the selected feature point, and ${}^{R}\mathbf{p}_{c} \in \Re^{2}$ and ${}^{R}\mathbf{p}_{o} \in \Re^{2}$ be the position of the camera and the object respectively from a fixed reference frame. The camera is modeled by the mapping $\sigma : \Re^{2} \times \Re^{2} \to \Re^{2}$ as given in equation 4-1. Since camera moves parallel to the target plane at a constant distance *z*, the actual model is given by

$$\boldsymbol{\eta} = \sigma({}^{R}\mathbf{p}_{c}, {}^{R}\mathbf{p}_{o}) = \frac{f}{z}\begin{bmatrix} \cos q & \sin q \\ -\sin q & \cos q \end{bmatrix}\left[{}^{R}\mathbf{p}_{o} - {}^{R}\mathbf{p}_{c}\right]$$

Here q is the sum of the joint angles q_1 and q_2, f is the focal length and z is the target depth from camera frame. It is assumed that camera completely recovers the spatial configuration of the object from visual information. The gain of the vision system is thus taken as unity. The inverse of the camera model σ^{-1} can map the image space information to the Cartesian space.

Since camera is generally not associated with any dynamics, its gain is considered an identity matrix. Camera is normally attached to a vision system which processes an image to extract target position and orientation information with respect to camera coordinate system. The feedback is delayed due to image acquisition and processing time. We assume that there is a constant delay of 100ms, no greater than the frame rate. The motion blur effects are ignored as we assume that there is sufficient scene illumination in order to maintain image brightness for short exposures.

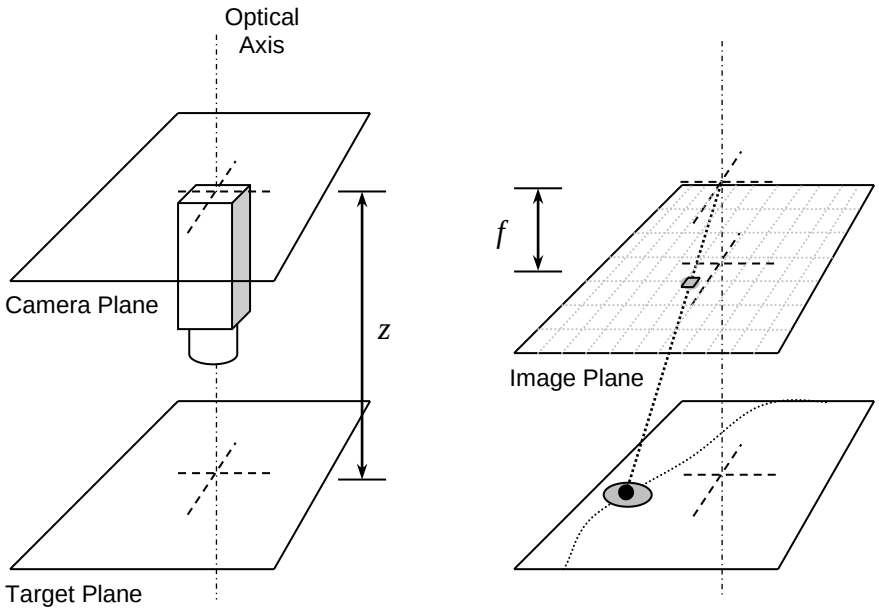


Figure 5-3: Orientation of camera, image and target frames

5.4 Control Architecture

We considered a two level hierarchical control scheme with a top-level multirate controller passing commands to the joint-level controller (Figure 5-4). Since control architecture is hierarchical and control law accepts Cartesian space error signal, it is a position-based dynamic look-and-move architecture, as described in section 3.3 . It is assumed that each joint axis is decoupled; hence it is possible to design the controller for each joint axis independently.

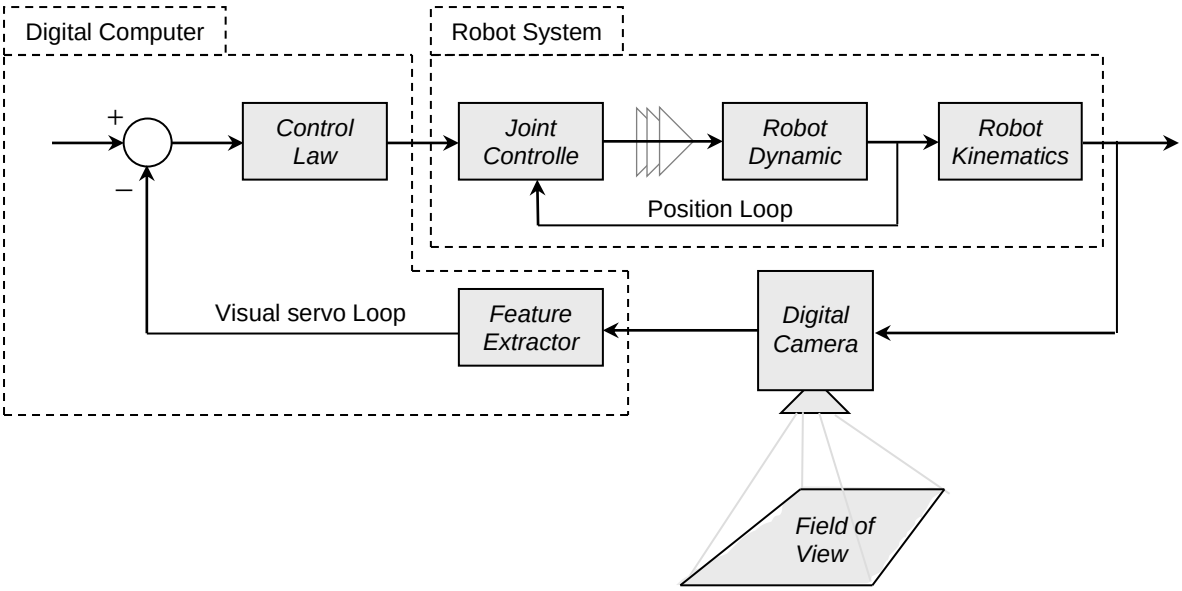


Figure 5-4: Visual servo control architecture

The inner loop workspace position controller is shown in (Figure 5-4). The transfer function from the joint-space acceleration command ($\ddot{q}$) to the joint-space position (q) is simply considered as a double integrator. With q_{ref} , the reference joint position, as input to the inner loop, a proportional-derivative control scheme is devised. Since the inner loop runs much faster than the outer vision loop, it is safe to consider the inner-loop as continuous-time plant model for outer vision loop. The plant model is given by

$$\mathbf{P}(s) = \frac{K_d s + K_p}{s^2 + K_d s + K_p}\mathbf{I}_2$$

Where K_p and K_d are the proportional and derivative gains of the PD-controller. We consider $K_p = 150$ and $K_d = 50$.

In order to maintain lesser simulation times, frame rate T_f of camera is assumed to be 100ms (instead of usual 25-40ms) and a joint sensor rate T_c of 10ms. A single-rate discrete-time controller is designed using conventional pole placement design philosophy. The multirate controller and disturbance rejection observer are designed using techniques mentioned in Chapter 4.

5.5 Performance Metric

There is lack of any agreed performance measures to compare the temporal performance of various visual servo systems [8]. Closed-loop system performance is usually measured in terms of bandwidth, however high bandwidth can lead to a system that is sensitive to noise and unmodeled dynamics.

Commonly used performance metrics for visual servoing are settling time, overshoot and polynomial signal following errors are appropriate. An appropriate performance metric for comparing different control approaches for fixation or tracking tasks is to evaluate the quality of the tracking motion in terms of image plane error [8], [24]. The RMS error over the simulation time interval with *N* sample times is given by

$$\lambda = \sqrt{\frac{\sum_{j=1}^{N} \left({}^{i}\mathbf{p}[j] - {}^{i}\mathbf{p}_{d}[j] \right)^{2}}{N}}$$

where ${}^{i}\mathbf{p}[j]$ represent the position of centroid of object in image at j^{th} sampling instant, while ${}^{i}\mathbf{p}_{d}[j]$ is the desired object centroid position.

Though, a feedback system which is able to settle accurately over a static target, it may well show poor ability to follow a sinusoid [2], [8]. It is thus chosen as an appropriate target motion with which to evaluate the control performance. For fixation control, a sinusoid is particularly challenging since it bears persistent acceleration and clearly reveals the phase error.

Chapter 6

Simulations and Results

The 2-link robot position control model implemented in SIMULINK using Robotics Toolbox is shown in Appendix E. The 2nd order, continuous time model for single joint is given in state-space representation as follow

$$\mathbf{A}=\begin{bmatrix}-50 & -150\\ 1 & 0\end{bmatrix},\qquad \mathbf{B}=\begin{bmatrix}1\\ 0\end{bmatrix},\qquad \mathbf{C}=\begin{bmatrix}50 & 150\end{bmatrix},\qquad \mathbf{D}=0 \quad \textbf{(6-1)}$$

We utilize single-rate and multirate control in visual servo control model (given in Appendix E) in turn for comparison. Sinusoid target path is especially considered, as it is known that a feedback system may show poor ability to track a sinusoid even if it is able to settle accurately over a static target. A circular path is also considered for the comparison.

6.1 Single-rate Control

The above single-axis model is discretized at 0.1sec for single-rate controller design. The state-space representation for the discrete-time controller with sampling time $T_c = T_f = 0.1\,\text{sec}$, is given by

$$\mathbf{\Phi}=\begin{bmatrix}1.5 & -0.5\\ 1 & 0\end{bmatrix},\qquad \mathbf{\Gamma}=\begin{bmatrix}1\\ 0\end{bmatrix},\qquad \mathbf{H}=\begin{bmatrix}0.048 & 0.03\end{bmatrix} \quad \textbf{(6-2)}$$

The step response of the controlled system (combined (6-1) and (6-2)) is shown in Figure 6-1. This single-rate control law is used for individual axis control. Since initial

Cartesian-space position of end-effector is (0.6858, 0), an initial state condition for *x*-axis control is determined.

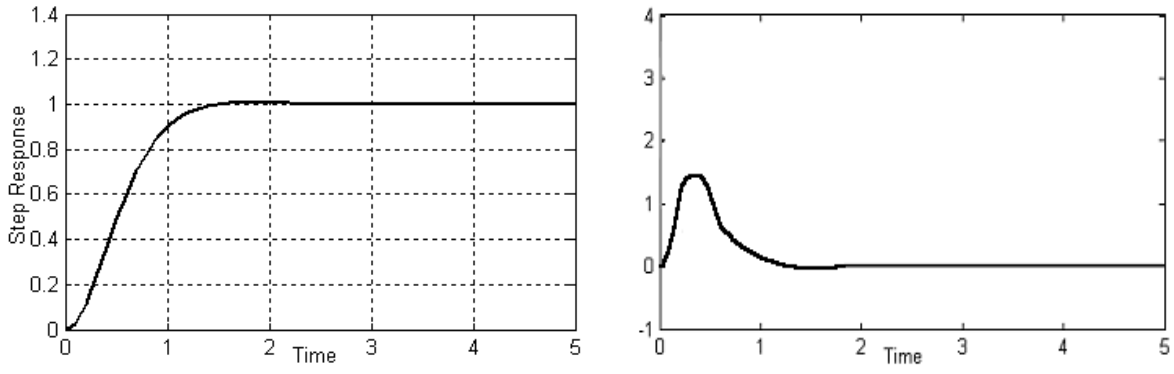


Figure 6-1: Step response and control-input of single-rate system given in (6-2)

6.1.1 Sinusoidal Path Tracking

The image space error is measured in terms of position of the centroid (the selected feature) in the image. The feature error in horizontal and vertical image coordinates is measured. The error in both axes is given in figure below.

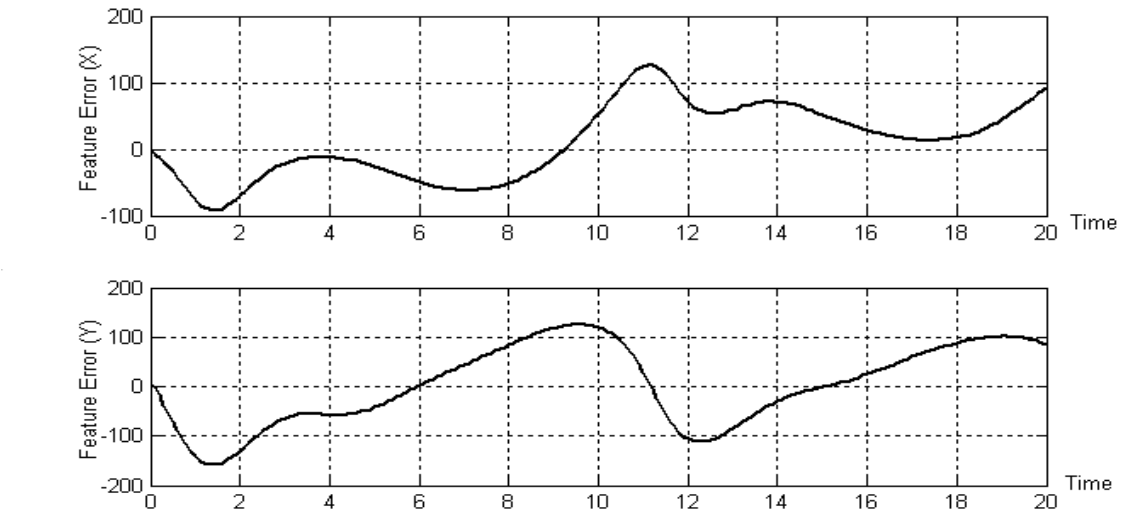


Figure 6-2: Image space error (in pixels) in position of object centroid.

The performance metric λ (given in Section 5.5) is evaluated on each axis. Following values are obtained

$$\lambda_x = 55.9524 \text{ pixels}$$

$$\lambda_y = 80.2418 \text{ pixels}$$

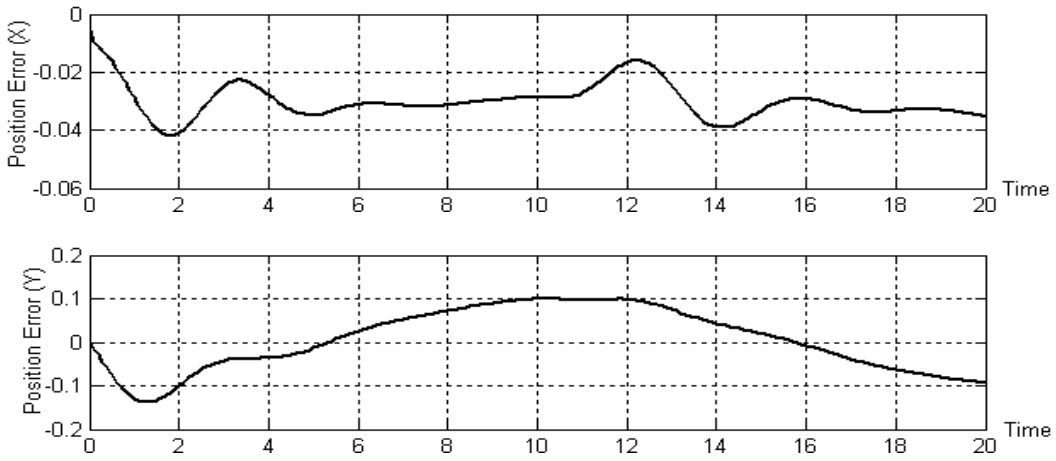


Figure 6-3: Actual Cartesian-space position error (in meters) b/w camera and object

6.1.2 Circular Path Tracking

In case of circular object movement, the feature and Cartesian space errors in both the axes are shown below. Values of performance metric are

$$\lambda_x = 52.6310 \text{ pixels}$$

$$\lambda_y = 87.4241 \text{ pixels}$$

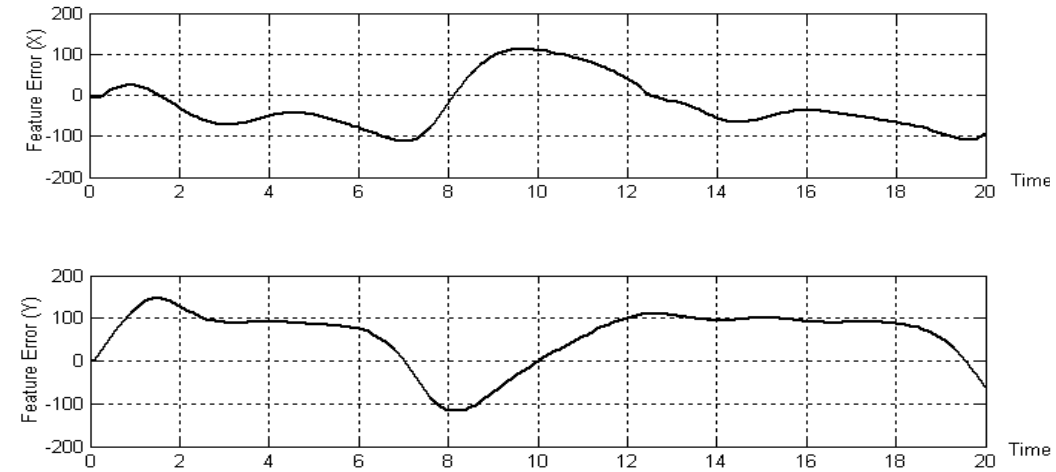


Figure 6-4: Image space error (in pixels) in position of object centroid

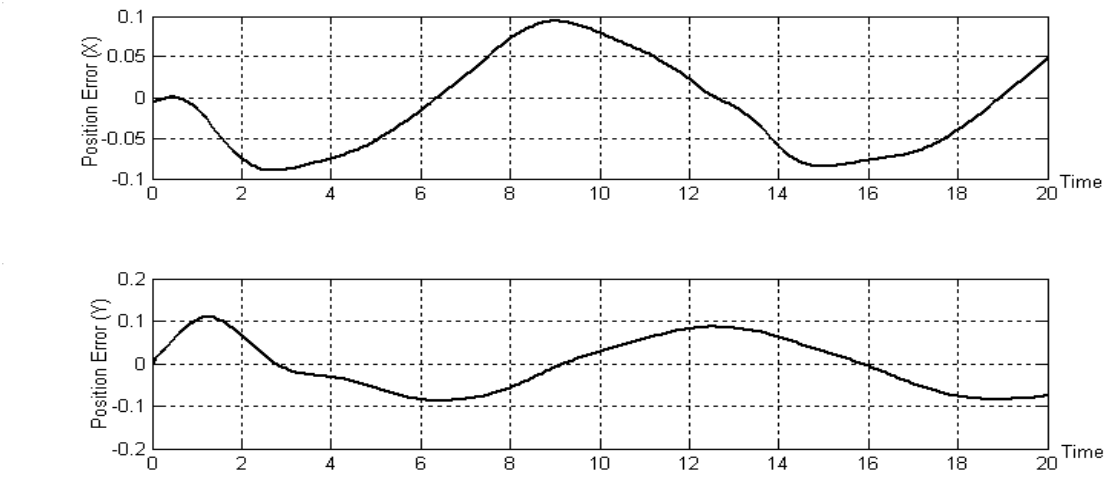


Figure 6-5: Actual Cartesian-space position error (in meters) b/w camera and object

6.2 Multirate Control

The continuous-time single-axis model is discretized at 0.01sec for multirate controller design with input multiplicity m = 10. A state feedback controller $\mathbf{K}$ and a current state observer gain $\mathbf{L}$ is designed first. Based on the formulation developed in (4-13) and (4-18), the lifted multirate plant model is given by

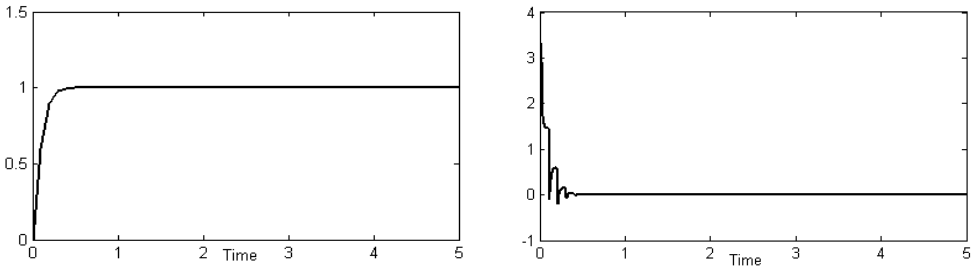
$$\underline{\mathbf{\Phi}} = \begin{bmatrix} -0.434 & -2.4655 \\ 0.0164 & 0.7784 \end{bmatrix}$$

$$\underline{\mathbf{\Gamma}} = \begin{bmatrix} -0.0004 & -0.0004 & -0.0003 & -0.0001 & 0.0002 & 0.0007 & 0.0014 & 0.0027 & 0.0047 & 0.0078 \\ 0.0002 & 0.0002 & 0.0002 & 0.0002 & 0.0002 & 0.0002 & 0.0002 & 0.0001 & 0.0001 & 0 \end{bmatrix}$$

The state feedback gain matrix is given by

$$\underline{\mathbf{K}} = \begin{bmatrix} -36.5806 & -116.8691 \\ -37.6015 & -120.1263 \\ -38.5539 & -123.1616 \\ -39.3747 & -125.7720 \\ -39.9626 & -127.6311 \\ -40.1552 & -128.2155 \\ -39.6922 & -126.6869 \\ -38.1573 & -121.7041 \\ -34.8851 & -111.1232 \\ -28.8119 & -91.5183 \end{bmatrix} \qquad \textbf{(6-3)}$$

It is obvious from (6-3) that the multirate controller is time varying. The step response and the corresponding control input is shown below



(a). Step response (b). Control input

Figure 6-6: Step response and control input for multirate control system

As obvious from the Figure 6-6, the step response is much quicker for multirate control. However the control usage is much high and the control signal is highly overshooting. This particular characteristic of multirate control is discussed in [11], [12], [18], [20], [22].

6.2.1 Sinusoidal Path Tracking

For the feature errors given in following figures, the performance metric in multirate case is computed as

$$\lambda_x = 23.6524 \text{ pixels}$$

$$\lambda_y = 40.9818 \text{ pixels}$$

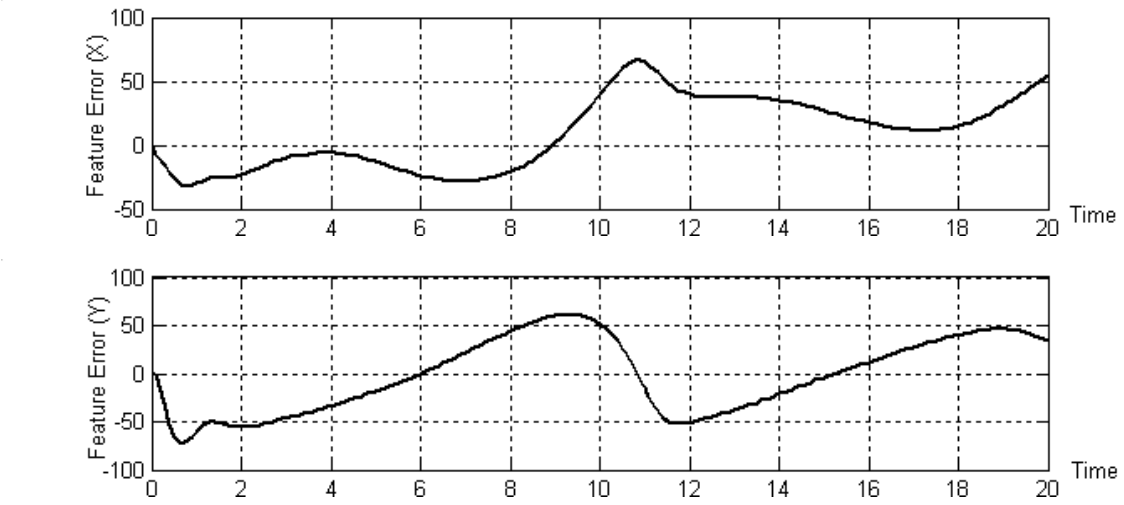


Figure 6-7: Image space error (in pixels) in position of object centroid

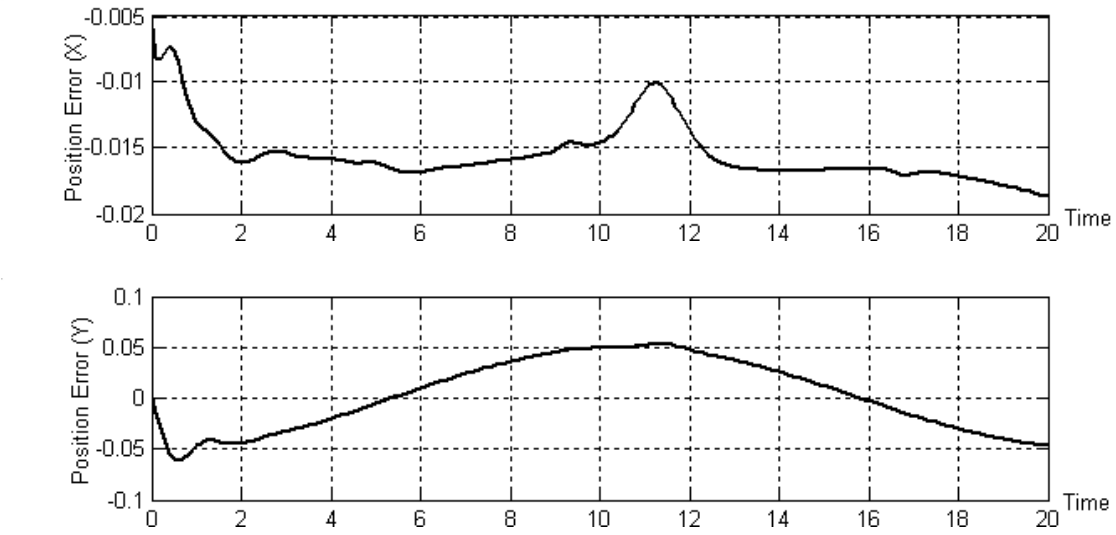


Figure 6-8: Actual Cartesian-space position error (in meters) b/w camera and object

6.2.2 Circular Path Tracking

In this case, the performance metric evaluates as

$$\lambda_x = 29.5274 \text{ pixels}$$

$$\lambda_y = 43.9781 \text{ pixels}$$

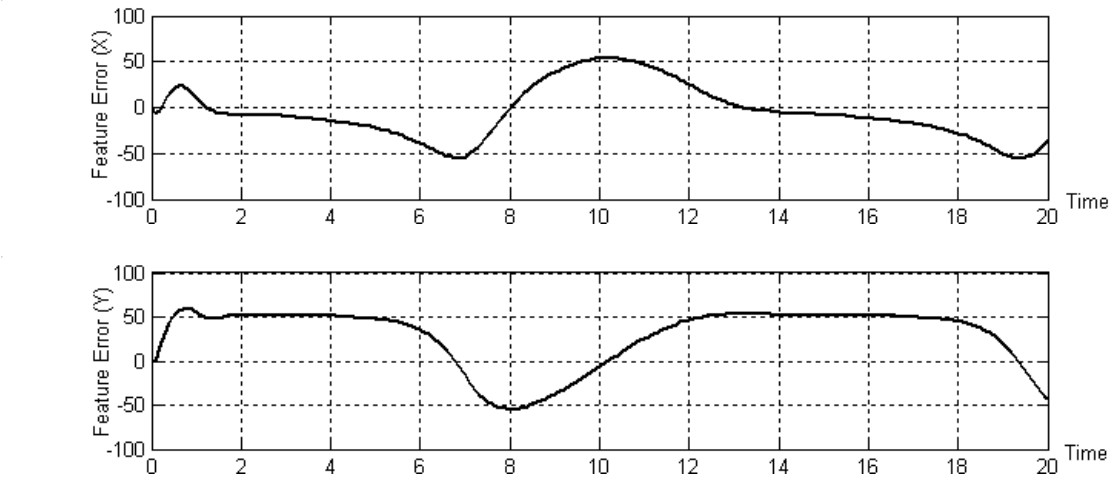


Figure 6-9: Image space error (in pixels) in position of object centroid

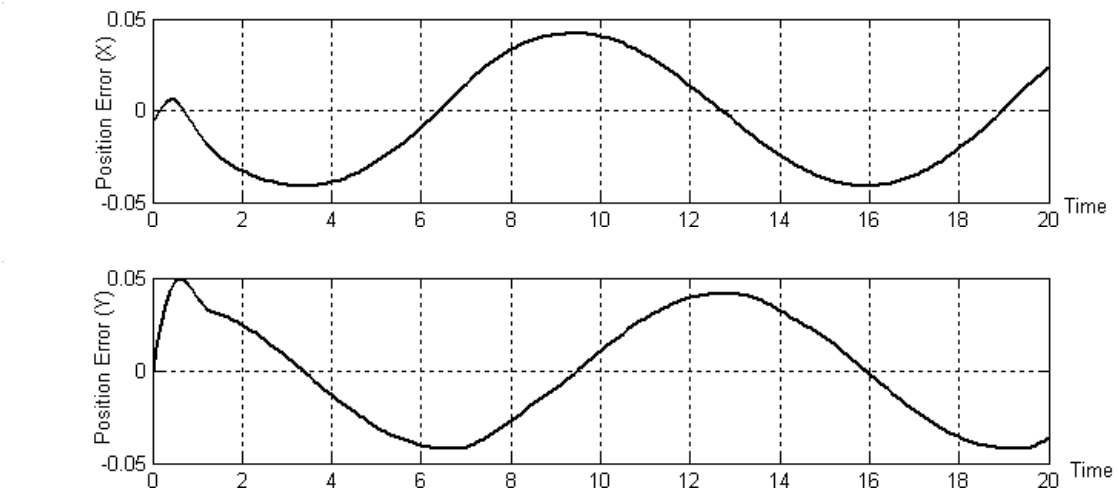


Figure 6-10: Actual Cartesian-space position error (in meters) b/w camera and object

6.3 Comparison with Other Approaches

Several other approaches to improve visual tracking performance while visual servoing has been proposed [8]. Since most robot controllers present the abstraction of a robot as a position-controlled device, majority of reported visual servo systems are based on the approach in which the visual servo is built on top of an underlying position control of the manipulator.

In [8], the comparison of image plane error for various visual feedback compensators has been presented, where in all cases the target motion is a sinusoid. The error in case of simple PID control is somewhat greater than 100 pixels, which is also demonstrated in our simulation results for single-rate PD control (see Figure 6-2). Utilizing target state estimation in a feedforward fashion makes an improvement to simple PID control [25][26]. The error reduces to nearly 50 pixels. These results are obtained without utilizing and adaptive or optimal control strategy, which can further improve the tracking performance [28][29].

Our multirate strategy compares in performance with simple feedforward control, yet never utilizes any of the target state estimation techniques. This fact figures out great potential for further improving tracking performance by making use of target position and velocity estimation methods.

6.4 Comments

It can be observed that the corresponding error curves are almost similar in both single-rate and multirate control. This is because the multirate design formulation is based on the discrete state-matching with a faster controller initially designed to synthesize lifted controller and plant representations. Also, the error rise is much steeper for about initial 1 second. This is due to the fact that the multirate control input signal is much higher in magnitude and oscillates rapidly. Though, the error grows well at start, but the multirate compensator tries to alleviate the error during intersample period i.e. between measurement sampling periods.

Note that the performance improvement in case of multirate is almost 50 % of the single-rate control performance. Though the system model was very crude, and not all the fine details of the system are captured in the simulation, still it can be well conjectured that multirate control approach can result in improved performance (if not as much as 50%) in most of the visual servoing tasks. Better calibration methods, improved vision system, and better state estimation techniques can fairly contribute towards further improvement.

Chapter 7

Concluding Remarks

7.1 Conclusion

In this thesis, we have studied the multirate controller, which updates the control output faster than the measurement sampling frequency by a factor of *m*, the input multiplicity. The problem of 2D visual servoing is considered. A conventional single-rate control scheme is also utilized for comparison.

We have utilized a special multirate design strategy based on discrete state matching. This multirate control approach delivered better tracking performance while visually tracking moving objects. However we observe that when such a multirate controller is used, the control input becomes oscillatory as compared to the single-rate controller. While oscillations of the control input may not be desirable in practice, the multirate controller may still be able to keep the closed loop system stable for a given measurement sampling period. The oscillation is due to the use of a zero order hold in discretizing the continuous time controller, the output of which is inherently discontinuous. However in practical implementation of multirate control, the control signal should be well examined, and the control parameters be thoroughly tuned as it may raise the actuators to its saturation limit or even cause damage to the system. Better understanding of system dynamics can contribute towards achieving maximum advantage out of a multirate strategy.

7.2 Future Directions

Because of the lack of availability of real robot system parameters, this work remained restricted to visual servoing using 2DOF planar robot. These ideas can be extended to a robot system with more degrees of freedom and with more complex servoing problem.

We have not utilized Kalman or $\alpha - \beta$ filters for target state estimation, since objective was to explore the tracking performance of raw single-rate and multirate control strategies. These filters applied in feedforward fashion can further improve tracking performance.

Nearly all reported stereo-vision based systems used fixed cameras. Almost all end-effector mounted experiments used monocular apparatus. However there seems no reason that a stereo-vision setup cannot be mounted on the end-effector.

We have used position-based visual servo architecture. Image-based architecture can also be utilized for comparison

Kalman filter has been accepted as a standard tool for state estimation in visual servoing. Not much work has been done for use of H_∞ filter in robotic and computer vision. They superiority of H_∞ filter over other estimation techniques is theoretically convincing since model uncertainties can be handled more adequately.

References

[1] Tongwen Chen, Bruce A. Francis (1995), "Optimal Sampled-Data Control Systems", Springer, London.

[2] Karl J. Åström, Björn Wittenmark (1997), "Computer-Controlled Systems: Theory and Design", 3rd ed. Prentice Hall, Englewood Cliffs, N.J.

[3] G. F. Franklin, J. D. Powell, M. L. Workman (2002), "Digital Control of Dynamic Systems", 3rd ed. Pearson Education Inc., India.

[4] Hiroshi Fujimoto (2000), *General Framework of Multirate Sampling Control and Applications to Motion Control Systems*, PhD Dissertation, Department of Electrical Engineering, University of Tokyo.

[5] Yuping Gu (2000), *Multi-rate Digital Control and Signal Processing: Theory and Applications to Motion Control Systems*, PhD Dissertation, Department of Mechanical Engineering, University of California at Berkley.

[6] M. Berg, N. Amit, J.D. Powell (1988), *Multirate Digital Control System Design*, IEEE Trans. on Automatic Control, vol. 33, no. 12, pp. 1139-1150.

[7] Jiandong Wang, Tongwen Chen, Biao Huang (2003), *Multirate Sampled Data Systems: Computing Fast-Rate Models*, Journal of Process Control, Elsevier Science.

[8] Seth Hutchinson, Greg Hager, Peter Corke (1996), *A Tutorial on Visual Servo Control*, IEEE Transactions on Robotics and Automation, vol. 12, no. 5.

[9] J. T. Feddema, O. Mitchell (1989), *Vision-Guided Servoing with Feature-Based Trajectory Generation*, IEEE Trans. on Robotics and Automation, vol. 5, no. 5.

[10] Hiroshi Fujimoto, Yoichi Hori (2001), *Visual Servoing Based on Intersample Disturbance Rejection by Multirate Sampling Control - Time Delay Compensation and Experimental Verification*, proceedings of 40th IEEE Conference on Decision and Control – CDC' 01, pp. 334--339, Orlando.

[11] Seung-Hi Lee, Chung Choo Chung, Sang-Min Suh (2003), *Multirate Digital Control for High Track Density Magnetic Disk Drives*, IEEE Transactions on Magnetics, vol. 39, no. 2.

[12] M. F. Sagfors, H. T. Toivonen, B. Lennartson (2000), *State-space Solution to the Periodic Multirate H_∞ Control Problem: A Lifting Approach*, IEEE Trans. on Automatic Control, vol. 45, no. 12, pp. 2345-2350.

[13] Douglas P. Glasson (1983), *Development and Applications of Multirate Digital Control*, IEEE Control Systems Magazine, November.

[14] J. Sklansky, J. R. Ragazzini (1955), *Analysis of Errors in Sampled-data Feedback Systems*, AIEE Transactions, vol. 74, part 11, pp. 65-71.

[15] G. M. Kranc (1957), *Input Output Analysis of Multirate Feedback Systems*, IRE Trans. on Automatic Control, vol. 3, pp. 21-28.

[16] R.E. Kalman, J. Bertram (1959), *A Unified Approach to the Theory of Sampling Systems*, J. Franklin Institute, vol. 267, pp. 405 – 436.

[17] Carlos Mario Vélez S., *References - Control with Non-Conventional Sampling*, www.control-systems.net, Department of Basic Sciences, EAFIT University, Medellin, Colombia.

[18] T. Chen, L. Qiu (1994), *H_∞ Design of General Multirate Sampled-Data Control Systems*, Automatica, vol. 30, pp. 1139–1152.

[19] David G. Meyer (1990), ***A*** *New Class of Shift-Varying Operators, Their Shift-Invariant Equivalents, and Multirate Digital Systems*, IEEE Trans. Automatic Control, vol. 35, no. 4, pp. 429-433.

[20] Sauro Longhi (1994), *Structural Properties of Multirate Sampled-Data Systems*, IEEE Trans. Automatic Control, vol. 39, no. 3, pp. 692-696

[21] P. P. Khargonekar, Kameshwar Poolla, Allen Tannenbaum (1985), *Robust Control of Linear Time-Invariant Plants Using Periodic Compensation*, IEEE Trans. on Automatic Control, vol. 30, no. 11, pp. 1088-1096.

[22] T. Hagiwara, M. Araki (1988), *Design of a Stable Feedback Controller Based on the Multirate Sampling of the Plant Output*, IEEE Trans. Automatic Control, vol. 33, no. 9, pp. 812-819.

[23] L. Weiss, A. Sanderson, C. Neuman (1987), *Dynamic Sensor-based Control of Robots with Visual Feedback*, IEEE Journal of Robotics and Automation, vol. 3, no. 5.

[24] Peter Corke (1995), *Dynamic Issues in Robot Visual Servo Systems*, International Symposium on Robotics Research ISRR' 95, pp 488--498, Herrsching, Germany.

[25] Peter Corke, M. C. Good (1993), *Controller Design for High Performance Visual Servoing*, Proceedings 12th World Congress International Federation of Automatic Control – IFAC' 93, Sydney, vol. 9, pp. 395-398.

[26] Peter Corke (1993), *Experiments in High Performance Robotic Visual Servoing*, proceedings International Symposium on Experimental Robotics – ISER' 93, Kyoto, Japan.

[27] S. Chroust, João P. Barreto, H. Araujo, M. Vincze (2001), *Comparison of Control Structures for Visual Servoing*, ICAR' 01 - IEEE International Conference on Advanced Robotics, Budapest, Hungary.

[28] N. P. Papanikolopoulos, P. K. Khosla (1993), *Adaptive Robotic Visual Tracking: Theory and Experiments*, IEEE Trans. on Automatic Control, vol. 38, no. 3, pp. 429-444.

[29] N. P. Papanikolopoulos, P. K. Khosla, T. Kanade (1993), *Visual Tracking of a Moving Target by a Camera Mounted on a Robot: A Combination of Control and Vision*, IEEE Trans. on Robotics and Automation, vol. 9, no. 1, pp. 14-35.

[30] João P. Barreto, Jorge Batista, Helder Araújo (2000), *Control Issues to Improve Visual Control of Motion: Application in Active Tracking of Moving Targets*, Proceeding of 6th International Workshop on Advanced Motion Control–AMC, Nagoya, Japan.

[31] João P. Barreto, Jorge Batista, Helder Araújo (2000), *Model Predictive Control to Improve Visual Control of Motion: Applications in Active Tracking of Moving Targets*, ICPR'00 – 15th IEEE International Conference on Pattern Recognition, Barcelona, Spain, p. 4732.

[32] Peter K. Allen, A. Timcenko, B. Yoshimi, P. Michelmen (1993), *Automated Tracking and Grasping of a Moving Object with a Robotic Hand-Eye System*, IEEE Trans. On Robotics and Automation, vol. 9, no. 2, pp. 152-165.

[33] J.P. Urban, J.L. Buessler and J. Gresser (1997), *Neural Networks for visual servoing in robotics*, Technical Report no. EEA-TROP-TR-97-05, Equip EEA-Groupe TROP, Univrsite de Haute-Alsace, France.

[34] S. Niwa, T. Masuda, Y. Sezaki (1999), *Kalman filter with time-variable gain for a multisensor fusion system*, IEEE/SICE/RSJ Int. Conf. on Multisensor Fusion and Integration of Intelligent Systems, Taipei, pp 56-61.

[35] Hiroshi Fujimoto (2003), "*Visual Servoing of 6 DOF Manipulator by Multirate Control with Depth Identification*", proceedings of 42nd IEEE Conf. on Decision and Control, Maui, Hawaii, pp. 5408-5413.

[36] Mahadevamurty Nemani, T. C. Tsao, S. Hutchinson (1994), *Multi-rate analysis and design of visual feedback digital servo control system*, ASME Journal of Dynamic Systems, Measurement and Control, vol. 116, pp 45-55.

[37] Peter I. Corke (1996), *A Robotics Toolbox for MATLAB*, IEEE Robotics and Automation Magazine, vol. 3, no. 1, pp. 24-32.

[38] K. Hashimoto, H. Kimura (1995), *Visual Servoing with Non-Linear Observer*, IEEE International Conference on Robotics and Automation, pp. 484-489.

[39] K. S. Fu, R. C. Gonzalez, C. S. G. Lee (1987), “Robotics: Control, Sensing, Vision and Intelligence”, McGraw Hill Book Co., Singapore.

Appendix A

Controllability and Observability of Lifted System

Definitions

The concepts of controllability and observability deal with the ability of a control system to measure and control the states of a given system. A state is said to be controllable if there is some piecewise continuous (for continuous) or constant (for discrete time systems) control signal $u(t)$ which is capable of changing the state from any given initial value to any desired final value in a finite length of time. If the state variable cannot be influenced by the control inputs u, that state is said to be uncontrollable. If all states are controllable then the system is said to be controllable. Similarly, a system is said to be observable if every initial state $\mathbf{x}(0)$ can be determined by observing the system output y over some finite time period (for continuous time systems) or finite number of samples (for discrete time systems).

Consider an n^{th} order SISO discrete-time system

$$\begin{aligned} \mathbf{x}[k+1] &= \mathbf{\Phi}\,\mathbf{x}[k] + \mathbf{\Gamma}\,u[k] \\ y[k] &= \mathbf{H}\,\mathbf{x}[k] \end{aligned} \tag{A-1}$$

The system is controllable if the following condition holds

$$\text{rank}\left[\mathbf{\Gamma} \quad \mathbf{\Phi\Gamma} \quad \mathbf{\Phi}^2\mathbf{\Gamma} \quad \ldots \quad \mathbf{\Phi}^{n-1}\mathbf{\Gamma}\right] = n \tag{A-2}$$

The system is observable if the following condition holds

$$\text{rank}\left[\mathbf{H} \quad \mathbf{H\Phi} \quad \mathbf{H\Phi}^2 \quad \ldots \quad \mathbf{H\Phi}^{n-1}\right]^T = n \tag{A-3}$$

Proof

Consider the following multirate system, as given in (4-9)

$$\begin{aligned} \mathbf{x}[i+1,0] &= \underline{\mathbf{\Phi}}\,\mathbf{x}[i,0] + \underline{\mathbf{\Gamma}}\,\underline{\mathbf{u}}[i,0] \\ y[i,0] &= \underline{\mathbf{H}}\,\mathbf{x}[i,0] \end{aligned} \tag{A-4}$$

where $$\left[\begin{array}{c|c} \underline{\mathbf{\Phi}} & \underline{\mathbf{\Gamma}} \\ \hline \underline{\mathbf{H}} & \mathbf{O} \end{array}\right] \Rightarrow \left[\begin{array}{c|ccc} \mathbf{\Phi}^m & \mathbf{\Phi}^{m-1}\mathbf{\Gamma} & \mathbf{\Phi}^{m-2}\mathbf{\Gamma} & \cdots \quad \mathbf{\Gamma} \\ \hline \mathbf{H} & \multicolumn{3}{c}{\mathbf{O}} \end{array}\right]$$

It is known that the triplet $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{H})$ is controllable and observable. The controllability matrix $\mathbf{P}_c$ for given multirate system is

$$\mathbf{P}_c = \left[\underline{\mathbf{\Gamma}} \quad \underline{\mathbf{\Phi}}\,\underline{\mathbf{\Gamma}} \quad \underline{\mathbf{\Phi}}^2\underline{\mathbf{\Gamma}} \quad \ldots \quad \underline{\mathbf{\Phi}}^{n-1}\underline{\mathbf{\Gamma}}\right]$$

For a SISO system, $\underline{\mathbf{\Phi}}$ is an $n \times n$ and $\underline{\mathbf{\Gamma}}$ is an $n \times m$ matrix. Hence the dimensions of $\mathbf{P}_c$ are $n \times nm$. From the definition of the controllability matrix of (A-1) given in (A-2), it is clear that $\mathbf{P}_c$ is composed of same columns as that in (A-2), but in reverse order and in multiples of $\mathbf{\Phi}$. The number of linearly independent columns is still n. Thus the rank of $\mathbf{P}_c$ is n, and the lifter model (A-4) is controllable under given conditions.

Now, the observability $\mathbf{O}_b$ matrix of (A-4) is given by

$$\mathbf{O}_b = \left[\underline{\mathbf{H}} \quad \underline{\mathbf{H}}\,\underline{\mathbf{\Phi}} \quad \underline{\mathbf{H}}\,\underline{\mathbf{\Phi}}^2 \quad \ldots \quad \underline{\mathbf{H}}\,\underline{\mathbf{\Phi}}^{n-1}\right]^T$$

where $\underline{\mathbf{H}}$ is $1 \times n$ vector. $\mathbf{O}_b$ is thus an $n \times n$ matrix. The number of linearly independent columns is still n. Thus the rank of $\mathbf{O}_b$ is also n, and the lifter model (A-4) is observable under given conditions.

Appendix B

Planar 2-Link Robot: Kinematics and Dynamics

Robot Kinematics

Kinematics refers to the geometric relationship between the motion of the robot in Joint Space and the motion of the end-effector in Task Space without consideration of the forces that produce the motion.

The *Forward Kinematics* Problem is to determine the mapping from Configuration Space to Task Space which gives the end-effector pose in terms of the joint configuration. The forward kinematics is determined by the physical geometry of the robot. *Inverse Kinematics* Problem is to determine the inverse of this mapping, i.e. the joint configuration as a function of the end-effector pose. The forward kinematics map is many-to-one, meaning that several Joint Space configurations may give rise to the same end-effector pose. This means that the forward kinematics always has a unique pose for each configuration, while the inverse kinematics has multiple solutions, in general.
A simple forward kinematic model for 2-link planar robot is given by

$$x = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2)$$
$$y = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2)$$

where l_1 and l_2 are the link lengths and θ_1 and θ_2 are the joint angles.

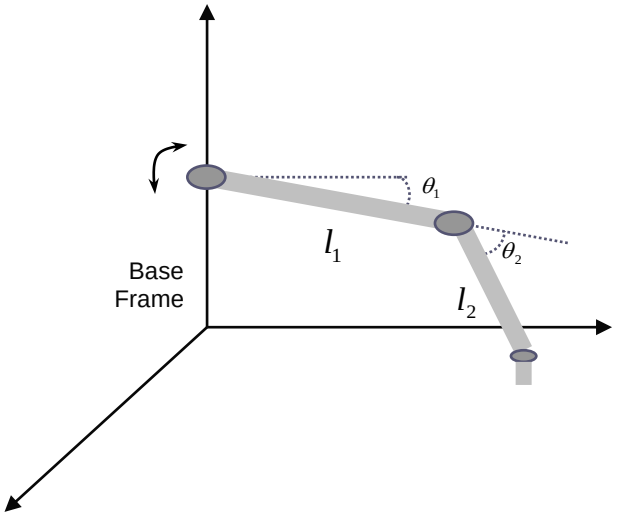


Figure: Two-link planar robot in RR configuration

Robot Dynamics

Robot arm dynamic deals with the mathematical formulation of the equations of robot arm motion. The dynamic performance of the manipulator directly depends on the efficiency of the control algorithms and the dynamic model of the manipulator.

The two major dynamic formulations are the Lagrange-Euler (L – E) formulation and Newton-Euler (N – E) formulations. L – E equations are computation intensive and difficult to utilize for real time control purposes unless they are simplified. N – E formulatioins are recursive and have lesser computational complexity.

The equations of motion for a two-link manipulator are given as

$$\boldsymbol{\tau}(t) = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}}(t) + \mathbf{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{C}(\boldsymbol{\theta})$$

Where $\boldsymbol{\tau}(t) = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$, $\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$, $\dot{\boldsymbol{\theta}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$, $\ddot{\boldsymbol{\theta}} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$

$$\mathbf{M}(\boldsymbol{\theta}) = \begin{bmatrix} \frac{1}{3}m_1 l_1^2 + m_2 l_1^2 + \frac{1}{3}m_2 l_2^2 + m_2 l_1 l_2 C_2 & \frac{1}{3}m_2 l_2^2 + \frac{1}{2}m_2 l_1 l_2 C_2 \\ \frac{1}{3}m_2 l_2^2 + \frac{1}{2}m_2 l_1 l_2 C_2 & \frac{1}{3}m_2 l_2^2 \end{bmatrix}, \quad \textit{with} \quad \begin{aligned} C_i &= \cos\theta_i \\ S_i &= \sin\theta_i \\ C_{ij} &= \cos(\theta_i + \theta_j) \end{aligned}$$

$$\mathbf{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \begin{bmatrix} -m_2 l_1 l_2 S_2 \dot{\theta}_1 \dot{\theta}_2 - \frac{1}{2}m_2 l_1 l_2 S_2 \dot{\theta}_2^2 \\ \frac{1}{2}m_2 l_1 l_2 S_2 \dot{\theta}_1^2 \end{bmatrix}$$

$$\mathbf{C}(\boldsymbol{\theta}) = \begin{bmatrix} \frac{1}{2}m_1 g l_1 C_1 + m_2 g l_1 C_1 + \frac{1}{2}m_2 g l_2 C_{12} \\ \frac{1}{2}m_2 g l_2 C_{12} \end{bmatrix}$$

Here $\mathbf{M}(\boldsymbol{\theta})$ represent the inertial term, $\mathbf{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ represent the *Coriolis* and centrifugal term, and $\mathbf{C}(\boldsymbol{\theta})$ represent the gravity term. m_1 and m_2 are the mass of the links and g is the acceleration due to gravity.

Appendix C

Robotics Toolbox

The Robotics Toolbox release 7.1 (April 2002) is used for the simulation of robot system. It is developed by Peter I. Corke at CSIRO and is a copyright of CSIRO Manufacturing Science and Technology.

This Toolbox provides many functions that are useful in robotics, such as robot forward and inverse kinematics and dynamics, trajectory generation etc. The Toolbox is useful for simulation as well as for analyzing results from experiments with real robots.

The Robotics Toolbox is freely available from the MathWorks FTP server ftp.mathworks.com or from http://www.cat.csiro.au/cmst/staff/pic/robot

Appendix D

Simulation Parameters

Robot Parameters

Table below gives the parameters for robot arm used to create a 'robot object' using Robotics Toolbox.

Manipulator Parameter	**Values**	
	Link-1	***Link-2***
Link Parameter		
Link twist angle (α in radians)	0	0
Link length (A in meters)	0.3810	0.3048
Link rotation angle (θ in radians)	0	0
Link offset distance (D in meters)	0	0
Joint type (σ) 0 for revolute, 1 for prismatic	0	0
Mass of the link (m in kg)	1.345	0.920
Moment of Inertia Tensor $= \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$ (I in kg.m^2)	$\begin{bmatrix} 9.529\times10^{-5} & 0 & 0 \\ 0 & 0.01627 & 0 \\ 0 & 0 & 0.01627 \end{bmatrix}$	$\begin{bmatrix} 5.573\times10^{-5} & 0 & 0 \\ 0 & 0.00712 & 0 \\ 0 & 0 & 0.00712 \end{bmatrix}$

Motor Parameters

Armature inertia (J_m)	0	0
Reduction gear ratio (*G*)	10	10
Viscous friction (*b*)	1.016×10^{-4}	1.016×10^{-4}
Coulomb friction (T_c+)	3.12×10^{-4}	2.92×10^{-4}
Coulomb friction (T_c-)	3.69×10^{-4}	2.49×10^{-4}

These values are taken from PUMA560 robot example given in Robotics Toolbox. Values are not exact, but slightly changed to match chosen link length and mass.

Vision System Parameters

Table below gives the parameters for the camera. The values are after [35].

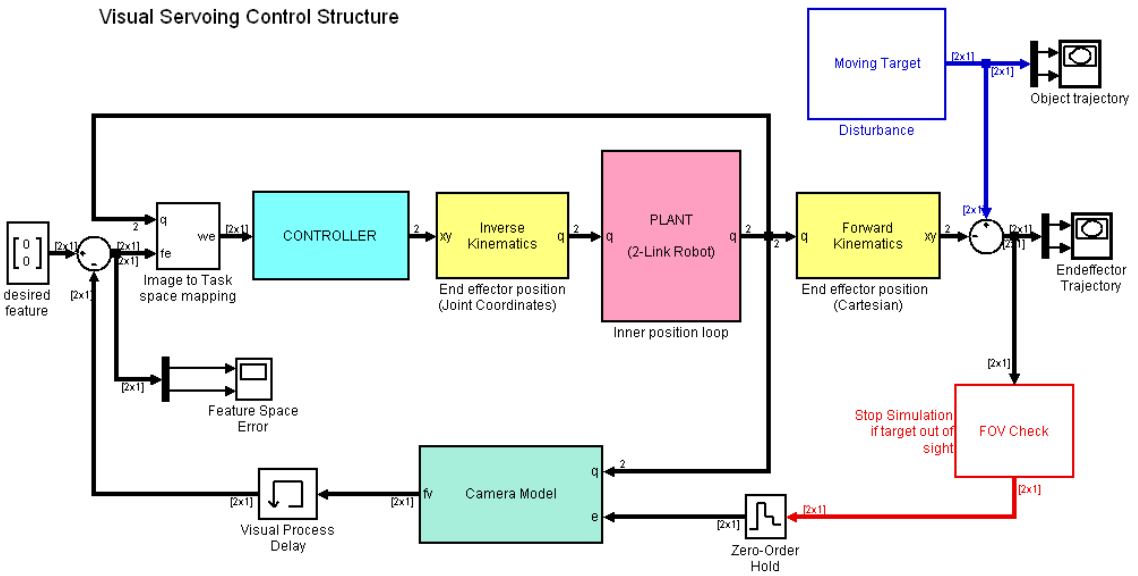
Image plane dimension (in pixels)	512×512
Focal length (in millimeters)	8
Pixel pitch (in pixels/meter)	80,000
Center point	(256, 256)

Appendix E

SIMULINK Models

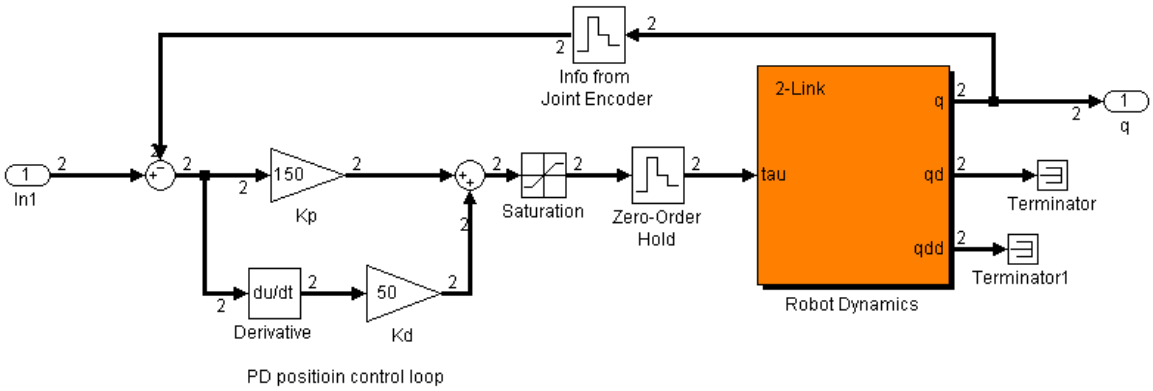
Main Model

The main SIMULINK model for simulation is shown below. The numbers on the connecting lines represent the signal dimension. The visual servo architecture is dynamic look-and-move based because of the control hierarchy and position-based because the controller decides on basis of Cartesian space error.

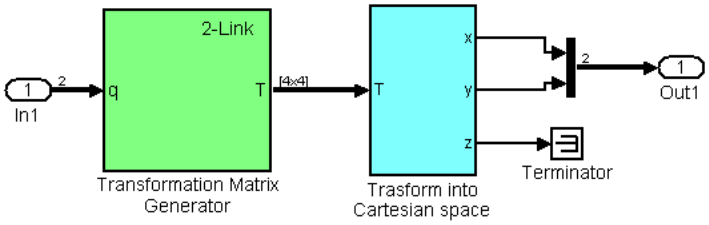


Plant Model

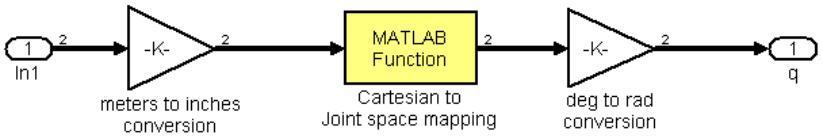
The plant model represent the 2-link robot object implemented in Robotics Toolbox along with proportional-derivative position control.



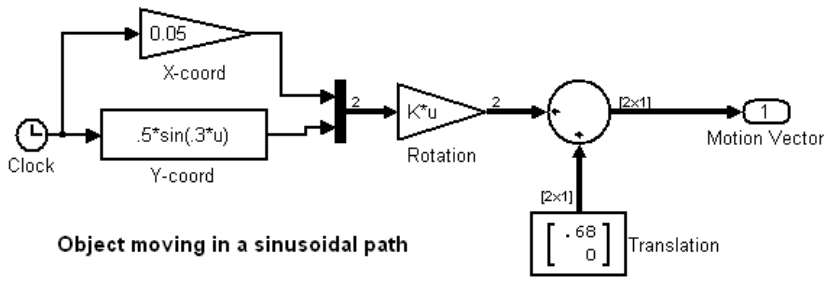
Forward Kinematics Block



Inverse Kinematics Block



Moving Target Block



FOV Check Block