

# 作業空間の非干渉化に基づいたビジュアルサーボ系の新しい設計法

高橋 啓\*, 坂東 信尚, 堀 洋一 (東京大学)

## A Novel Design Method of Visual Servoing System Based on Workspace Decoupling Control

Kei Takahashi, Nobutaka Bando and Yoichi Hori(The University of Tokyo)

### Abstract

The visual servoing system is composed of an object and an eye-in-hand robot. The object is moving around the workspace and the robot is tracking the object by using a visual sensor mounted on the hand.

A novel design method of visual servoing system based on workspace decoupling control is proposed in this paper. Workspace decoupling control makes the whole system simpler, therefore state feedback based on pole assignment method can be applied. The proposed design method enables us to adjust transient response of camera position arbitrarily. The effectiveness of proposed method is shown through some simulation and experimental results.

キーワード: ビジュアルサーボ, 作業空間の非干渉制御, 状態フィードバック, 極配置

( Keywords: visual servoing, workspace decoupling control, state feedback, pole assignment )

### 1 はじめに

現在、様々な分野においてロボットが使用されているが、自律動作、あるいは未知の環境下での動作が近年益々要求されている。従来のロボットは、エンコーダ等内界センサからの信号のみに基づいて制御されているため、ロボットの機構モデルや外界環境が変化する、あるいは未知であるとそれに対応できず、タスクの実行に障害が生じる。

産業ロボット等、整備された環境で与えられた作業を繰り返す場合には、内界センサのみで十分であったが、災害地区や宇宙空間などで作業を行う極限ロボットや、病院や家庭で活躍するロボットにおいては、全くの未知な環境で、自律的に作業を行う必要があるため、外界の情報を検出し、制御に用いるための機構が必要となる。

しかし視覚情報を用いて、ロボットを操作する際には、画像解析に要するむだ時間が長いことが、視覚制御の実現を困難にしていた最大の要因であった。そのため初期の研究では、ロボットは静止した状態で撮像し、修正目標に向かって動作するという Look-and-Move (“見てから動く”) を繰り返す方法が用いられた [1]。

これは 画像処理と 3 次元情報の計算、それに基づく軌道計画には膨大な計算が必要なため、オフライン処理が必要である、との発想に基づいており、静的フィードバックと位置付けられる。その結果、視覚が観測される度にロボットの動きが中断され、動的応答が遅くなる。結果として、動く目標を追跡するなどのリアルタイムタスクは困難となっていた。

近年主として計算機性能の向上に伴って、アームを動作させながら連続的に視覚情報を獲得し、フィードバックするビジュアルサーボが研究されるようになった。近年の視覚フィードバックの研究では、動く物体の実時間トラッキングを目標とした視覚サーボ法が盛んである。これは作業環境を理解してからおもむるに動き出す”見てから動く”方法とは異なり、作業対象物

とアームの間の状態を観測しながら反射的に反応する”見ながら動く”方法であり、センサ情報の処理、軌道計画、モータ制御がすべて並列して動くという特徴を持つ [2]。

ビジュアルサーボは、目標値入力の違いにより位置ベース法と特徴ベース法の二つに大別される。特徴ベース法によるビジュアルサーボイングシステムでは、これまで主に画像ヤコビアンと呼ばれる、関節変位の差分から特徴量の変分への関係を用いて関節位置指令が生成されていた。しかし、画像ヤコビアンが非線形であるために系全体が非線形となり、系の解析的な設計が困難になってしまう欠点があった。

そこで本研究ではこの問題を解決すべく、マニピュレータに対する入力に関節変位を用いることなく、作業空間での目標軌道を採用することにより、系全体のシンプルかつ解析的な記述が可能な、新しい設計法を提案する。

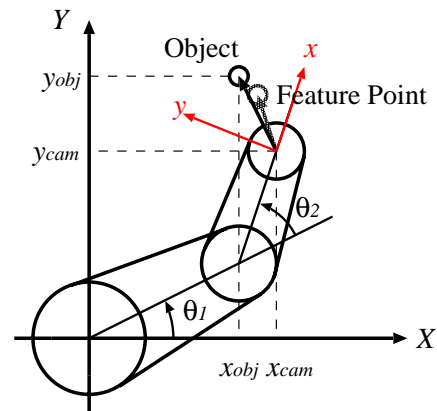


図 1: 2 軸 DD ロボットを用いたビジュアルサーボイングシステム

## 2 ビジュアルサーボ系の基礎的事項

### 2.1 ビジュアルサーボ系の構成

#### 2.1.1 システム全体の構成

本研究では、ビジュアルサーボ系として図1に示すような、先端にカメラが上向きに取り付けられた2軸DDロボットを用いた系を考える。系は二自由度であるため、目標物体とカメラの $z$ 方向の距離は不変かつ既知とする。制御目的はカメラ位置を目標物体位置に一致させ、リアルタイムで追従することである。ここで、カメラの位置を $s_{cam}$ 、物体の位置を $s_{obj}$ とする。

#### 2.1.2 カメラのモデルと特徴量

特徴量は、カメラ中心を原点にした物体の位置 $(X, Y)^T$ を画像平面に写像したものであり、

$$\begin{bmatrix} \xi_x \\ \xi_y \end{bmatrix} = \frac{f}{z} \begin{bmatrix} X \\ Y \end{bmatrix} \stackrel{\text{def}}{=} \iota^+(s_{cam}, s_{obj}) \quad (1)$$

と表される。 $f$ はカメラの焦点距離、 $z$ はカメラと目標物体間の $z$ 軸方向の距離である。 $\iota^+$ は $s_{cam}, s_{obj}$ からの写像であり、この写像によって表される $(x, y)^T$ を特徴点とする。本研究では特徴量は単一の特徴点で表され、このベクトルを特徴量ベクトル $\xi$ とする。

本研究のシステムにおける特徴量は、実座標に対するカメラ座標の回転角だけ $s_{obj} - s_{cam}$ を回転・拡大(縮小)した形になり、(2), (3)式で表される。

$$\xi = \begin{bmatrix} \xi_x \\ \xi_y \end{bmatrix} = \iota(\theta) \begin{bmatrix} x_{obj} - x_{cam} \\ y_{obj} - y_{cam} \end{bmatrix} \quad (2)$$

$$\iota(\theta) = \frac{f}{z} \begin{bmatrix} \cos(\theta_1 + \theta_2) & \sin(\theta_1 + \theta_2) \\ -\sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (3)$$

ただし $\iota(\theta)$ は $s_{obj} - s_{cam}$ から特徴量 $\xi$ への写像である。

### 2.2 ビジュアルサーボ系に関する研究の分類

#### 2.2.1 目標物体の運動モデルに関する分類

ビジュアルサーボ系に関する研究を目標物体の運動の取扱いに関して分類すると、以下ようになる。

1. 目標物体が静止している場合
2. 目標物体が運動する場合
  - (a) 目標物体の運動があらかじめわかっている場合
  - (b) 目標物体の運動が未知である場合

目標物体が静止しているとする場合は、目標物体の運動を考慮する必要はなく、画像処理のむだ時間や制御入力の調整などが問題となる。むだ時間に対して直接

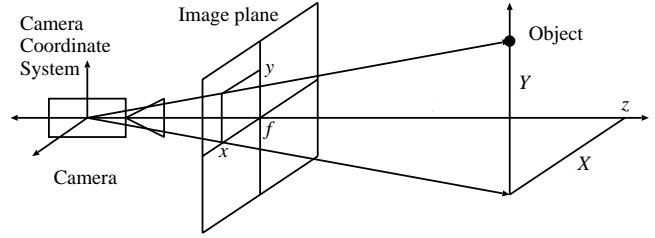


図2: 目標物体の撮像モデル

的にアプローチした研究としては、岡崎ら[3]のものなどが挙げられる。また、橋本、木村らはむだ時間そのものにはアプローチせず、系の安定性を向上させる手法[4][5]を提案している。

目標物体が運動し、その軌道が既知であるとする研究では、予め与えられた目標軌道とリアルタイムの画像情報を用いて、より正確に制御を行うことが目的となる。遠藤、田中らは、目標軌道全体の情報を利用した最適予測制御を用いることで、滑らかかつ正確な軌道追従を実現した[7]。全目標軌道が既知であれば、むだ時間に対する対処をとらなくとも望ましい制御が可能となる。

また、一般に目標物体の運動が未知である場合は、物体の運動に関する情報は全くのリアルタイムでしか得られないため、物体の運動が既知である場合と同様な精度で制御を行うことは難しい。任意の軌道を前提とした研究としては、非線形オブザーバを用いた手法[6]などがある。

#### 2.2.2 ビジュアルサーボ系におけるむだ時間と予測に関する分類

視覚フィードバック制御系では、特に目標物体が運動する場合において、視覚センサ及び画像処理による大きなむだ時間が大きく制御性能を劣化させ、これが問題となる場合が多い。むだ時間に対して直接的にアプローチする方法としては、物体の運動予測を行う方法[9][6]、むだ時間による位相遅れを直接的に補償する方法[3]がある。

#### 2.2.3 画像ヤコビアンを用いた従来手法

これまでに提案されている視覚フィードバック制御法のほとんどは、画像ヤコビアンの逆行列に基づくものである。画像ヤコビアンは式(4)のように表され、画像特徴量の差分とマニピュレータの関節角の差分との関係である。図3のように、画像ヤコビアンを逆行列を用いて、関節角変位の目標値を生成することができる。

$$J(\theta, s_{cam}) = \frac{\partial \xi}{\partial s_{cam}} \frac{\partial s_{cam}}{\partial \theta} \quad (4)$$

$$= \begin{bmatrix} \xi_y - \frac{f}{z} L_1 \sin(\theta_2) & \xi_y \\ -\xi_x - \frac{f}{z} L_1 \cos(\theta_2) - \frac{f}{z} L_2 & -\xi_x - \frac{f}{z} L_2 \end{bmatrix}$$

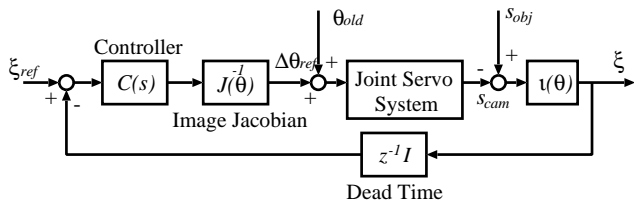


図 3: 画像ヤコビアンを用いた制御手法

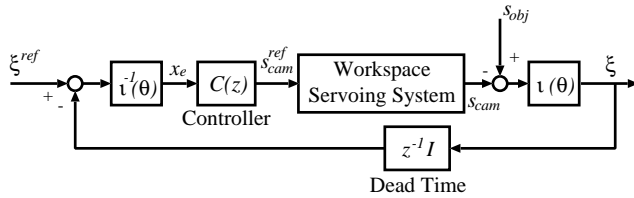


図 4: 作業空間に基づいた制御手法

本研究で対象とする系 (図 1) では画像ヤコビアンは (4) 式となる。画像特徴量  $\xi$  とマニピュレータの関節角  $\theta$  は観測可能なため画像ヤコビアンの実現は簡単であるが、画像ヤコビアンは非線形であるため、コントローラ的设计を解析的に行うのは困難である。また画像ヤコビアンは、画像空間上の特徴量から関節空間上の関節角指令を直接的に生成するため、一般的にタスクが記述される作業空間上で記述することはできない。

### 2.3 本研究の立場

画像ヤコビアンを用いる手法には上記のような問題点があるため、本研究では画像ヤコビアンを用いず、作業空間上で一元的に記述する制御手法を適用する。作業空間上でタスクを記述するためには、制御対象であるマニピュレータを作業空間で非干渉化することが必要である。その上で、図 4 のようにカメラモデルの逆行列である  $i^{-1}(\theta)$  を挿入することで、線形化が達成でき、併せてコントローラ  $C(z)$  の解析的な設計が可能となる。本研究においては、目標物体の運動は未知であるものとする。

また、本研究の目的は制御器の解析的な設計であり、線形予測などの簡単な運動予測では逆に系の安定性を損なうことがあるため、むだ時間に対する対処として 2.2.2 節で述べた目標物体の運動予測は、本研究では行わないものとする。

## 3 作業空間の非干渉化による ビジュアルサーボ系の線形化

### 3.1 作業空間の非干渉化

マニピュレータに作業空間で動作指令を与えるためには、作業空間において非干渉化する必要がある。本

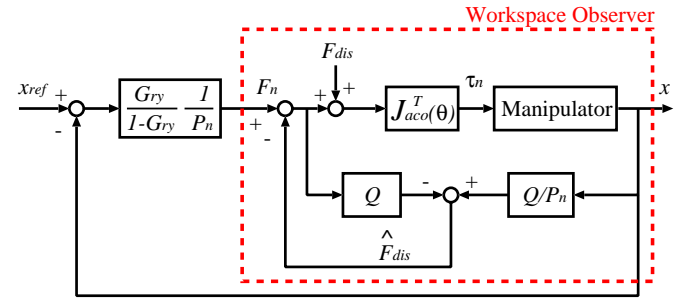


図 5: 作業空間オブザーバによる非干渉化

研究では、作業空間での外乱オブザーバに基づいて作業空間の非干渉化を実現する [8]。図 5 の点線枠内に示すように、作業空間において外乱オブザーバを構成する。ここで、 $Q$  および  $P_n$  は、以下のように定義されるローパスフィルタ、プラントのノミナルモデルである。

$$Q = \begin{bmatrix} \frac{g}{s+g} & 0 \\ & \ddots \\ 0 & \frac{g}{s+g} \end{bmatrix}, P_n = \begin{bmatrix} \frac{1}{M_n s^2} & 0 \\ & \ddots \\ 0 & \frac{1}{M_n s^2} \end{bmatrix} \quad (5)$$

このように  $P_n$  を対角行列とすることにより、作業空間の非干渉化が達成できる。また、作業空間の位置ベクトル  $x$  を (6) 式のように表すと、作業空間の速度ベクトル・加速度ベクトルはそれぞれ (7), (8) 式となる。ただし、 $J_{aco}$  はヤコビ行列である。また作業空間での力ベクトル  $F_n$  と関節空間でのトルクベクトル  $\tau_n$  との関係は (9) 式になる。

$$x = H(\theta) \quad (6)$$

$$\dot{x} = \frac{\partial H(\theta)}{\partial \theta} \dot{\theta} = J_{aco} \dot{\theta} \quad (7)$$

$$\ddot{x} = J_{aco} \ddot{\theta} + \dot{J}_{aco} \dot{\theta} \quad (8)$$

$$\tau_n = J_{aco}^T F_n \quad (9)$$

この関係を用いて、図 5 においては生成した力指令から各軸のトルク指令を生成するために、ヤコビ転置行列を用いている。

外乱オブザーバにより、 $Q$  におけるゲイン  $g$  が十分大きい場合には図 5 の点線枠内部分は  $P_n$  にノミナル化される。さらに、位置フィードバック部分のコントローラを

$$\frac{G_{ry}}{1 - G_{ry} P_n} \quad (10)$$

とすることにより、目標値応答が  $G_{ry}$  (希望目標値応答と呼ぶ) となる。このように外乱オブザーバを用いた制御系は、目標値追従特性と外乱抑圧特性が独立に設計できる、ロバストな位置制御系が実現される [10]。以下、位置制御系の目標値応答として  $G_{ry} = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2}$  を用いる。

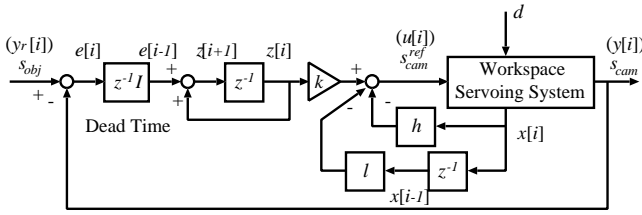


図 6: 状態フィードバック法

### 3.2 カメラの逆モデルによる系の線形化

視覚センサにより、画像特徴量ベクトル  $\xi$  は (3) 式に示したように、 $\theta_1 + \theta_2$  だけ回転し、拡大 (縮小) した形となる。制御系内にこの非線形な要素が、解析的な制御系設計にとってのもう一つの問題点である。この非線形要素を打ち消すために、カメラモデル ((3) 式) の逆行列である  $\iota^{-1}(\theta)$  をフィードバックループに挿入する (図 4)。マニピュレータの関節角である  $\theta$  はエンコーダによりリアルタイムで測定可能であり、 $f/z$  は実験によって事前に測定できるので、 $\iota^{-1}(\theta)$  の実現は簡単である。これにより、系の非線形要素は全て線形化される。

## 4 制御器の設計

### 4.1 極配置に基づいた状態フィードバック

作業空間で非干渉化された位置制御系の指令値に対する  $x, y$  各軸の特性は  $G_{ry}$  であるため、位置制御系を制御対象としてビジョンループの制御周期で離散化する。その結果得られた、離散時間制御対象を (11), (12) 式とする。

$$\mathbf{x}[i+1] = \mathbf{A}\mathbf{x}[i] + \mathbf{B}u[i] \quad (11)$$

$$y[i] = \mathbf{C}\mathbf{x}[i] \quad (12)$$

本研究では図 6 のような状態フィードバックにより、制御系設計を行う。制御系の入力为目标物体位置  $s_{obj}$  とし、出力をカメラ位置  $s_{cam}$  としてとらえる。また、この場合の制御系は目標物体位置を位置指令とするサーボ系であり、ステップ状の目標値を仮定して、閉ループ内に内部モデルを持つように、 $\frac{1}{z-1}$  を挿入する。

$$z[i+1] = z[i] + e[i-1] \quad (13)$$

$$= z[i] + y_r[i-1] - \mathbf{C}\mathbf{x}[i-1] \quad (14)$$

であるので、式 (11) と式 (14) をあわせた拡大系の状態方程式は

$$\begin{bmatrix} \mathbf{x}[i] \\ \mathbf{x}[i+1] \\ z[i+1] \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}[i-1] \\ \mathbf{x}[i] \\ z[i] \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \\ 0 \end{bmatrix} u[i] + \begin{bmatrix} \mathbf{0} \\ \mathbf{d} \\ y_r[i-1] \end{bmatrix} \quad (15)$$

と表される。このとき、

$$u[i] = \begin{bmatrix} -l & -h & k \end{bmatrix} \begin{bmatrix} \mathbf{x}[i-1] \\ \mathbf{x}[i] \\ z[i] \end{bmatrix} \quad (16)$$

の状態フィードバックを施すと、系全体は、

$$\begin{bmatrix} \mathbf{x}[i] \\ \mathbf{x}[i+1] \\ z[i+1] \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{B}l & \mathbf{A} - \mathbf{B}h & \mathbf{B}k \\ -\mathbf{C} & \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}[i-1] \\ \mathbf{x}[i] \\ z[i] \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{d} \\ y_r[i-1] \end{bmatrix} \quad (17)$$

となる。ただし  $h, l$  は、

$$h = \begin{bmatrix} h_1 & h_2 \end{bmatrix}, l = \begin{bmatrix} l_1 & l_2 \end{bmatrix} \quad (18)$$

で表されるゲイン行列である。また、

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{B}l & \mathbf{A} - \mathbf{B}h & \mathbf{B}k \\ -\mathbf{C} & \mathbf{0} & 1 \end{bmatrix} \quad (19)$$

をシステム行列と呼ぶ。拡大系の状態方程式 (15) は可制御ではなく、システム行列の全ての固有値をフィードバックゲインである  $k, h, l$  によって自由に操作できるわけではない。しかし操作できない固有値は  $z = 0$  にあるため、このシステムは可安定であるといえる。すなわち、極配置に基づいた状態フィードバックによってシステム行列の極を指定することにより、指令値である目標物体位置からの応答が自由に調節できる。また、極を全て零に設計することにより、有限整定サーボ系が実現される。

### 4.2 極の位置による応答の変化

前節に述べた状態フィードバックにより、ビジュアルサーボシステムの極を自在に配置できるようになる。本節では、離散系のシステム行列の固有値の配置により、応答がどのように変化するかを調べ、単純なモデルによるシミュレーションにより検討する。連続時間における  $s$  平面と離散時間における  $z$  平面との関係は、サンプリング周期  $T$  を用いて

$$z = e^{Ts} \quad (20)$$

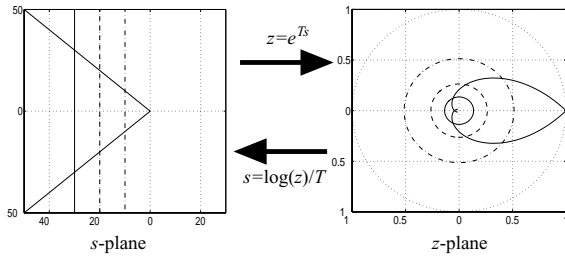


図 7:  $s$  平面と  $z$  平面との対応

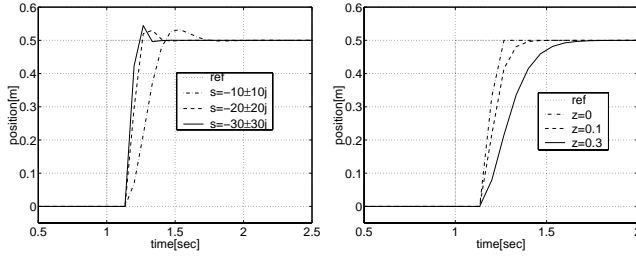


図 8:  $s = -10 \pm 10j, -20 \pm 20j, -30 \pm 30j$  に極を配置した場合のステップ応答

図 9:  $z$  平面実軸上に極を配置した場合のステップ応答

と表される。(20) 式より、 $s$  平面と  $z$  平面との対応は、図 7 のようになる。 $T = 66.6[\text{msec}]$  として、一次元の位置制御モデル  $G_{ry} = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2}$  をプラントとしたシミュレーションを行った。ただし、システム行列の操作できない極 ( $z = 0$ ) 以外の四つの極を、 $s$  平面上で

$$s = -10 \pm 10j, -10 \pm 10j \quad (21)$$

$$s = -20 \pm 20j, -20 \pm 20j \quad (22)$$

$$s = -30 \pm 30j, -30 \pm 30j \quad (23)$$

のように配置し、これを (20) 式によって  $z$  平面の極に変換した。ステップ位置指令に対する応答の変化を図 8 に示す。 $s$  平面状での応答と同様に、適度なオーバーシュートをもち、極の位置によって応答速度が変化していることが理解される。

さらに図 9 に、極を実軸上の 4 重根とし、 $z = 0$  (有限整定)、 $z = 0.1, 0.3$  と変化させた場合のステップ応答を示す。 $z$  平面で単位円内の実軸は、 $s$  平面での左半平面の実軸に対応するので、応答はオーバーシュートのない、臨界制動的なものとなっている。

## 5 実験による比較・検討

### 5.1 実験装置の概要

提案する設計手法の有効性を検証するため、2 軸 DD ロボットを用いた実験を行った。実験システム全体の構成図を図 10 に示す。実験では一定のサンプリング周期で計算を行うことや、またカメラが得た画像情報を PC にリアルタイムで受け渡すことにリアルタイム

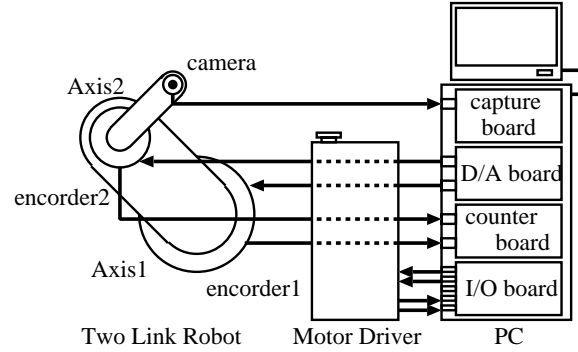


図 10: 実験システムの構成

性が要求されるために、OS として RT-Linux を用いた。目標物体は、ステッピングモータを用いて円盤を回転させ、その円盤の端部に印を付けてそれを追従させる構成とした。円盤は、2 軸 DD ロボットの上に組まれたやぐらに取り付けた。

ロボットの各パラメータは第 1 軸の長さが  $L_1 = 0.359[\text{m}]$ 、第 2 軸の長さが  $L_2 = 0.24[\text{m}]$ 、 $F = f/z = 5.5$  である。また、目標物体は半径  $5[\text{cm}]$  の円周軌道上を角速度  $0.98\pi[\text{rad/sec}]$  で等速円運動させるか、あるいは静止させるものとした。作業空間位置制御系の希望目標値応答  $G_{ry}$  は、 $x, y$  各軸で

$$G_{ry} = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2} \quad (24)$$

とし、 $\omega_n = 15[\text{rad/sec}]$  とした。また、 $Q$  のゲインは  $g = 150[\text{rad/sec}]$  とし、作業空間で非干渉化するサンプリング周期は  $1[\text{msec}]$  とした。ビジョンループのサンプリング周期は  $66.6[\text{msec}]$  とし、特徴量の算出までにはちょうど一周期のむだ時間を持たせた。

### 5.2 実験結果

四つの極を実軸上に 4 重根として配置した場合のカメラ軌道を図 11 に示す。凡例における数値は 4 重根を示す。極を  $0.1, 0.3, 0.4$  と  $z = 0$  から遠ざけるにつれ、応答のオーバーシュートは小さくなり、ゆっくりとした応答となっている。逆に極を  $z = 0$  に近付けるにつれて、速い応答になっていることがわかる。

しかし、極を全て零とした有限整定サーボ系では、オーバーシュートが大きくなりすぎ、振動的な応答となっている。

また、極を負の実軸上においたときには、系は安定性を失った。これは、極が  $z$  平面の負方向の実軸上にあるときには、収束が振動的となり、不安定になりやすいためと思われる。

図 12 に目標物体を等速円運動させた場合の  $X$  方向の位置、図 13 に画像特徴量の時間変化を示す。ただし目標とする画像特徴量は原点であるので、特徴量が小さいほど追従誤差は小さい。4 重根の極の場合、その位置が  $z$  軸上で単位円内ならば、安定性に問題はないことがわかる。また、図 11 においてオーバーシュートが少なく、かつ速い応答を示した、 $z = 0.2$  に 4 重

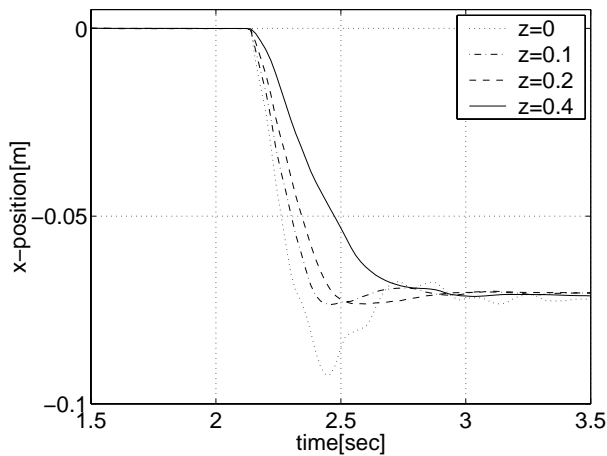


図 11: 目標物体が静止している場合のカメラ位置 ( $X$  方向) の時系列

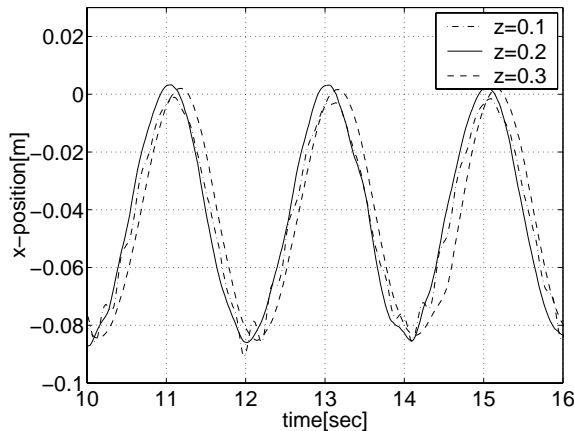


図 12: 目標物体が円運動している場合のカメラ位置  $x$  の時系列

根を置いた場合に最も追従誤差が小さいという結果が得られた。

## 6 まとめ

本論文では画像ヤコビアンを用いないビジュアルサーボ系の設計法として、プラントを作業空間において非干渉化し、極配置に基づいた状態フィードバック法を用いた設計法を提案した。この手法の大きな長所は設計の見通しの良さにある。また、単純化したモデルを用いたシミュレーションによって、極の配置で過渡応答を自由に变化させられることを示した。最後に、実験によってこの設計法の有効性を検証した。

今後の課題としては、物体の運動予測をも含めた際の、コントローラの設計法や、安定性に関する考察が挙げられる。

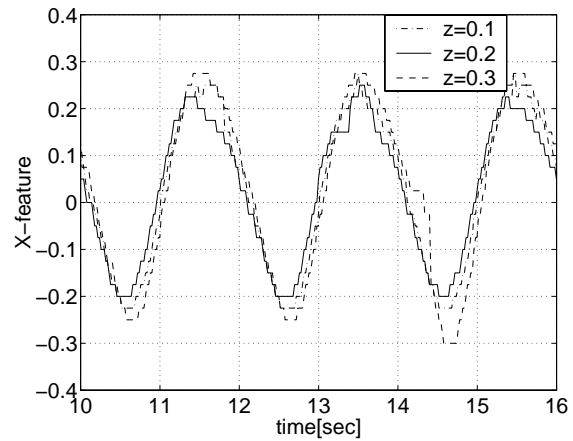


図 13: 目標物体が円運動している場合の画像特徴量  $\xi_x$  の時系列

## 参考文献

- [1] 石川正俊, "ビジュアルサーボイングの現状と将来", 計測と制御, Vol.40, No.9, pp.615-617, 2001.
- [2] 橋本浩一, "視覚フィードバック制御 - 静から動へ", システム/情報/制御, Vol.38, No.12, pp.21-27, 1994.
- [3] 岡崎俊太郎, 藤本博志, 堀洋一, "むだ時間を持つ視覚フィードバック制御系における高速でロバストな制御器の一提案", 電気学会産業計測制御研究会 IIC-00-18, pp.103-108, 2001.
- [4] 橋本浩一, 海老根巧, 木村英紀, "ビジュアルサーボイング-最適制御によるアプローチ-", 日本ロボット学会誌, Vol.12, No.5, pp.773-778, 1994.
- [5] 橋本浩一, 井上貴博, 木村英紀, "ビジュアルサーボイング-非線形制御アプローチ-", 日本ロボット学会誌, Vol.13, No.2, pp.263-269, 1995.
- [6] 橋本浩一, 木村英紀, "ビジュアルサーボイング-非線形オブザーバアプローチ-", 日本ロボット学会誌, Vol.13, No.7, pp.92-99, 1995.
- [7] 遠藤公誉, 田中弘一, 荒川賢一, 武川直樹, "予見制御による視覚軌道追従制御法", 日本ロボット学会誌, Vol.15, No.4, pp.85-92, 1997.
- [8] 村上俊之, 大西公平, "ロバスト制御に基づいた多自由度ロボットの安定性及び作業空間での非干渉制御に関する一考察", 電気学会論文誌 D, Vol.113, No.5, pp.639-646, 1993.
- [9] 猿田訓彦, 藤本博志, 堀洋一, "運動推定オブザーバによる運動項推定を用いたビジュアルサーボ", 電気学会産業計測制御研究会 IIC-01-13, 2001.
- [10] 梅野孝治, 堀洋一, "2 自由度ロバストサーボ系の構成とマニピュレータの運動制御への応用", 電気学会論文誌 D, Vol.110, No.11, pp.1163-1170, 1990.