2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Stability / Precision Improvement of 6-DoF Visual Servoing by Motion Feedforward Compensation and Experimental Evaluation

Wei Song and Mamoru Minami

***Abstract*— This paper deals with position-based 6-DoF visual servoing. With a common sense of feedback control, we stress that improvement of the dynamics of the sensing unit is important for a stable visual servoing. We propose a method to improve dynamics in visual recognition, with compensating the fictional motion of the target in the camera images based on kinematics of the manipulator, by extracting the real motion of the target. We named it as hand-eye motion feedforword (MFF) method. The enhanced dynamics of recognition gave further stability and precision to the total visual servoing system, evaluated by full 6-DoF servoing experiment using 7-link manipulator. The convergence time in step response was about 10[s] and precise visual servoing to a moving target object has been achieved.**

I. Introduction

Tasks in which visual information are used to direct a manipulator toward a target object are referred to visual servoing, as shown in Fig.1. Generally, visual servoing can be described as a feedback control as shown in Fig.2. The following things are well-known in a feedback control theory. Let dY denote the change of the output Y, it gives

$$\frac{dY}{Y} = \frac{1}{1+CSH}\frac{dS}{S}. \tag{1}$$

Usually $CSH >> 1$, the change of S will not affect the output a lot, which indicates that the influence from changing the dynamics of the system could be suppressed by the effect of feedback.

Let H be changed as dH, then the change of the output Y is

$$\frac{dY}{Y} = -\frac{CSH}{1+CSH}\frac{dH}{H}. \tag{2}$$

Giving $CSH >> 1$, we can get the following approximate expression

$$\frac{dY}{Y} \fallingdotseq -\frac{dH}{H}. \tag{3}$$

Eq. (3) indicates that the change of H will affect the output directly even with the high controller gain. This analysis displays the uncertainty and time delay of the dynamics of H affects the output dynamics directly more than the change of S, and it reduces the stability of visual servoing. Therefore, improvement of the dynamics of the sensing unit is essential for stable visual servoing. As shown in Fig.2, hand-eye motion disturbs recognition in H, and incorrect recognition will cause hand motion Y to be unstable, and the disturbed Y amplifies servoing error. This repeating

Wei Song and Mamoru Minami are with Department of Human and Artificial Intelligence Systems, Graduate School of Engineering, University of Fukui, Fukui, 910–8507, Japan { songwei, minami } @rc.his.fukui-u.ac.jp

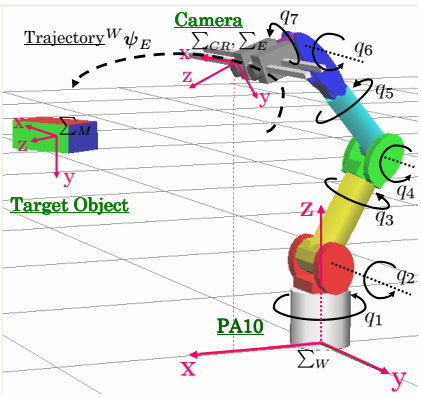


Fig. 1. Visual servo system of PA-10

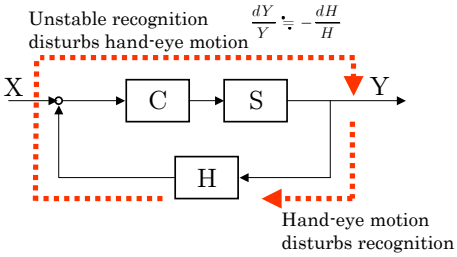


Fig. 2. Feedback system

in feedback loop may lead to dangerous unstable motion. Such an undesirable circulation is preferably cut down by improving the recognition dynamics to make the system be robust against the hand-eye motion.

However, research regarding the sensing dynamics for visual servoing has not been concentrated energetically so far. In [1], the authors used extended kalman filters to predict the target's pose in the real world. They considered from the view point of improving the ability of the servo controller, and did not pay attention to the recognition unit in the visual servoing system. In contrast, in this paper, we separate the target motion seen from the camera into two parts: one is the real motion from the target itself, the other is the fictional motion caused by the hand-eye camera's dynamical motion. We propose a method to compensate the fictional motion through the camera motion calculated by the kinematics of the manipulator and the observation of joint angles and angular velocities. We named it as hand-eye motion feedforword (MFF) method. Once the fictional motions is compensated during on-line recognition by MFF, it seems that the recognition were performed by using just fixed cameras in task space, then the recognition dynamics can be separated from the dynamics of the manipulator. Thus

TABLE I

POSITION-BASED AND IMAGE-BASED VISUAL SERVOING

	Advantage	Drawback
Position-based	Trajectory planning is done in an intuitive Cartesian coordinate space	Require a model of the target object
	There is a clear separation of the measurement problem and the control problem.	Require camera and robot calibration
	Familiar robot control design is used	
Image-based	do not require a model of the target object	difficult to do trajectory planning in the non-intuitive image plane,
	Best suited to planar motion where the plane is parallel to the image plane	difficult to non-planar motion where the plane is not parallel to the image plane
	Robust to camera and robot calibration errors	

the recognition becomes easier and the recognition dynamics can be improved.

Visual servoing can be classified into two major groups: position-based and image-based visual servoing [2]. Position-based visual servoing is to determine the object pose in Cartesian coordinate frame and lead to Cartesian robot motion planning, [3], [4]. On the other hand, in an image-based visual servoing, image features are measured in the 2-D image space, and the robot is controlled directly to servo the image features to a set of desired locations, [5], [6], without recognizing the target pose in 3-D space. The advantages and drawbacks of each visual servoing method have been discussed by a significant amount of researches, listing in Table I. Comparing image-based visual servoing with position-based visual servoing, the latter is more understandable, since the way of the visual servo is more like human-being, who positions perceived pose in Cartesian space during dynamical action like sports. This 3-D space perception does not exist in the image-based servoing, and thus position-based method suits to the MFF compensation.

We will show three experiments in this paper to evaluate the effectiveness of our system through full 6-DoF visual servoing experiments using 7-link manipulator. The first one is step response, in which the convergence time is about 10[s] that shows a good ability comparing with the other researches; the second is time-varying path control experiment; and the third is visual servoing to a moving target that is fixed on a mobile robot. Through these experiments, we will confirm the effectiveness of our system to enhance dynamics of recognition that gave further stability and precision to the total visual servoing system.

II. ON-LINE EVOLUTIONARY RECOGNITION

A. Background of 3-D Object Recognition

There is a variety of approaches for 3D target object's pose estimation, and they can be classified into three general categories: feature-based, appearance-based, and model-based.

Feature-based approaches use local features like points, line segments, edges, or regions to match against the incoming video to update the estimation pose. Feature-based techniques are naturally less sensitive to occlusions, as they are based on local correspondences. Some researches apply this method to head pose estimation based on tracking of small facial features like the corners of the eyes or mouth, [8]. Appearance-based (also template-based) approaches take the template as a whole. The image is compared with various templates to determine which one most closely matches the image, resulting in wasting time to recognize. In [9], the surface of the target 3D object is modeled by a set of small square patches, which needs a learning process to be determined from several key views. The third method is to use a model to search a target object in the image, and the model is composed based on how the target object can be seen in the input image [11], [12]. Our method is included in this category.

B. 3-D Model-based Matching

First, we give the definitions of coordinate systems used in this paper. The world coordinate frame is represented as Σ_W, the target coordinate frame is Σ_M, the end-effector coordinate frame is Σ_E and the camera coordinate frame is Σ_{CR}, as shown in Fig. 1. Here, the left camera is fixed parallel with the right one, so they are considered as one coordinate frame is Σ_{CR}. Σ_E is assumed the same as Σ_{CR} since the camera is mounted on the robot's end-effector.

We use a model-based matching method to recognize a target object in a 3-D searching area. A solid models is located in Σ_{CR}, its position and orientation are determined by six parameters, $\boldsymbol{\psi} = [\boldsymbol{r}^T, \boldsymbol{\epsilon}^T]^T$, where $\boldsymbol{r} = [x, y, z]^T$, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \epsilon_3]^T$. Here, the target's orientation is represented by unit quaternion [14], which has an advantage that can represent the orientation of a rigid body without singularities. The unit quaternion, viz. Euler parameters, defined as

$$\boldsymbol{Q} = \{\eta, \boldsymbol{\epsilon}\}, \tag{A.1}$$

where

$$\eta = cos\frac{\theta}{2}, \epsilon = sin\frac{\theta}{2}\boldsymbol{k}, \tag{A.2}$$

where $\boldsymbol{k}(\|\boldsymbol{k}\| = 1)$ is the rotation axis and θ is the rotation angle. η is called the scalar part of the quaternion while ϵ is called the vector part of the quaternion. They are constrained by

$$\eta^2 + \boldsymbol{\epsilon}^T\boldsymbol{\epsilon} = 1. \tag{4}$$

In (4) η can be calculated by $\boldsymbol{\epsilon}$, so we just use three parameters $\boldsymbol{\epsilon}$ to represent an orientation.

The left and right input images from the stereo cameras are directly matched by the left and right searching models, which are projected from 3-D model onto 2-D image plane.

The matching degree of the model to the target can be estimated by a fitness function $F(\boldsymbol{\psi})$ by using the color information of the target. Please refer to [13] for a detailed definition of $F(\boldsymbol{\psi})$. When the searching models fit to the target objects being imaged in the right and left images, $F(\boldsymbol{\psi})$ gives the maximum value. Therefore the 3-D object's position/orientation measurement problem can be converted to a searching problem of $\boldsymbol{\psi}$ that maximizes $F(\boldsymbol{\psi})$. We solve this optimization problem by 1-step GA method that will be explained in the next section.

C. GA-based On-line Recognition

The theoretically optimal pose $\boldsymbol{\psi}^{max}(t)$ that gives the highest peak of $F(\boldsymbol{\psi}(t))$ is defined as

$$\boldsymbol{\psi}^{max}(t) = \{\boldsymbol{\psi}(t) \mid \max_{\boldsymbol{\psi}\in\boldsymbol{L}} F(\boldsymbol{\psi}(t))\}, \tag{5}$$

where $\boldsymbol{L}$ represents 6-DoF searching area of $x, y, z, \epsilon_1, \epsilon_2, \epsilon_3$.

Here we use GA to search $\boldsymbol{\psi}^{max}(t)$. The individual of GA is defined as $\boldsymbol{\psi}_{i,j}(t)$, which means the i-th gene ($i = 1, 2, \cdots, p$) in the j-th generation. Denote $\boldsymbol{\psi}_{ga}^{max}(t)$ as the highest peak in GA process,

$$\boldsymbol{\psi}_{ga}^{max}(t) = \{\boldsymbol{\psi}_{i,j}(t) \mid \max_{\boldsymbol{\psi}_{i,j}\in\boldsymbol{L}} F(\boldsymbol{\psi}_{i,j}(t))\}. \tag{6}$$

In fact we cannot always guarantee the best individual of GA $\boldsymbol{\psi}_{ga}^{max}(t)$ correspond to the theoretically optimal pose $\boldsymbol{\psi}^{max}(t)$, because the number of GA's individuals is limited. The difference of $\boldsymbol{\psi}^{max}(t)$ and $\boldsymbol{\psi}_{ga}^{max}(t)$ is denoted as

$$\delta\boldsymbol{\psi}(t) = \boldsymbol{\psi}^{max}(t) - \boldsymbol{\psi}_{ga}^{max}(t). \tag{7}$$

And the difference of $F(\boldsymbol{\psi}^{max}(t))$ and $F(\boldsymbol{\psi}_{ga}^{max}(t))$ is denoted as

$$\Delta F(\delta\boldsymbol{\psi}(t)) = F(\boldsymbol{\psi}^{max}(t)) - F(\boldsymbol{\psi}_{ga}^{max}(t)) \geq 0. \tag{8}$$

Here, we present two assumptions.

[Assumption 1]: Assuming that $F(\boldsymbol{\psi}(t))$ distribution satisfies $\Delta F(\delta\boldsymbol{\psi}(t)) = 0$ if and only if $\delta\boldsymbol{\psi}(t) = 0$.

[Assumption 2]: Assuming that $\dot{F}(\boldsymbol{\psi}_{ga}^{max}(t)) > \dot{F}(\boldsymbol{\psi}^{max}(t))$, which indicates that the convergence speed to the traget in the dynamic images should be faster than the changing speed of the dynamic $F(\boldsymbol{\psi}(t))$ distribution as time t varying.

From [Assumption 2], we have

$$\Delta\dot{F}(\delta\boldsymbol{\psi}(t)) = \dot{F}(\boldsymbol{\psi}^{max}(t)) - \dot{F}(\boldsymbol{\psi}_{ga}^{max}(t)) < 0. \tag{9}$$

These two assumptions depend on some factors such as object's shape, object's speed, definition of $F(\boldsymbol{\psi}(t))$, parameters of GA and viewpoint for observing. We could set such an environment to satisfy or close to the above two assumptions. When above two assumptions are satisfied, (8) and (9) will be satisfied, then $\Delta F(\delta\boldsymbol{\psi}(t))$ is so-called Lyapunov function. That means $\Delta F(\delta\boldsymbol{\psi}(t))$ will be gradually decreased to 0. Thus, from the above definitions, we have $\delta\boldsymbol{\psi}(t) \to 0$, which means gradual stability in searching space $\boldsymbol{L}$, that is

$$\boldsymbol{\psi}_{ga}^{max}(t) \to \boldsymbol{\psi}^{max}(t), (t \to \infty) \tag{10}$$

Let t_ϵ denotes a convergence time, then

$$|\delta\boldsymbol{\psi}(t)| = |\boldsymbol{\psi}^{max}(t) - \boldsymbol{\psi}_{ga}^{max}(t)| \leq \epsilon, \quad (\epsilon > 0, t > t_\epsilon) \tag{11}$$

In (11), ϵ is the tolerable extent that can be considered as a observing error. Thus, it is possible to realize real-time optimization, because $\boldsymbol{\psi}_{ga}^{max}(t)$ is or near to the theoretically optimal $\boldsymbol{\psi}^{max}(t)$ after t_ϵ.

The above discussion is under the condition of varying time. Here, when we consider evolution time of each generation of GA denoted by Δt. The GA's evolving process is described as

$$\boldsymbol{\psi}_{i,j}(t) \xrightarrow{\text{evolve}} \boldsymbol{\psi}_{i,j}(t + \Delta t). \tag{12}$$

Obviously, this evolution time Δt will be possible to generate somewhat bad influence. If we assume that this bad influence on $\delta\boldsymbol{\psi}(t)$ can be described as

$$|\delta\boldsymbol{\psi}(t)| \leq \epsilon', \quad (\epsilon' > \epsilon > 0), \tag{13}$$

then, it can be considered Δt can manage real-time optimal solution. In (13), ϵ' is also tolerable extent as a observing error and it is somewhat larger than ϵ. Since the GA process is executed only one time to output the semi-optimal $\boldsymbol{\psi}_{ga}^{max}(t)$, we named this on-line recognition method as "1-step GA".

We have confirmed that the above time-variant optimization problem could be solved by 1-step GA through several experiments [7]. $\boldsymbol{\psi}_{ga}^{max}(t)$ will be output as the measurement result in each generation to control the robot manipulator. We define

$$\hat{\boldsymbol{\psi}}(t) = \boldsymbol{\psi}_{ga}^{max}(t), \hat{\boldsymbol{\psi}} = [\hat{x}, \hat{y}, \hat{z}, \hat{\epsilon_1}, \hat{\epsilon_2}, \hat{\epsilon_3}]^T. \tag{14}$$

III. Hand-eye Motion Compensation

The motion of the target seen from the camera will be affected by both the motion of the target in the real world and the motion of the camera in a eye-in-hand system. Here we describe such a relationship by a mathematical function, which can distinguish these two motions.

A. Analysis of target's motion in Σ_E

Take Σ_W as the reference frame. Denote the vector from O_W (the origin of Σ_W) to O_{CR} expressed in Σ_W as ${}^W\boldsymbol{r}_{CR}$, the vector from O_W to O_M expressed in Σ_W as ${}^W\boldsymbol{r}_M$, and the vector from Σ_{CR} to Σ_M expressed in Σ_{CR} as ${}^{CR}\boldsymbol{r}_M$. The following relations hold:

$${}^{CR}\boldsymbol{r}_M = {}^{CR}\boldsymbol{R}_W(\boldsymbol{q})({}^W\boldsymbol{r}_M - {}^W\boldsymbol{r}_{CR}(\boldsymbol{q})), \tag{15}$$

where ${}^{CR}\boldsymbol{R}_W$ is a rotation matrix determined by $\boldsymbol{q}$. Differentiating (15) with respect to time

$$\begin{aligned} {}^{CR}\dot{\boldsymbol{r}}_M = {}^{CR}\boldsymbol{R}_W(\boldsymbol{q})({}^W\dot{\boldsymbol{r}}_M - {}^W\dot{\boldsymbol{r}}_{CR}) + \boldsymbol{S}({}^{CR}\boldsymbol{\omega}_W) \\ {}^{CR}\boldsymbol{R}_W(\boldsymbol{q})({}^W\boldsymbol{r}_M - {}^W\boldsymbol{r}_{CR}(\boldsymbol{q})). \end{aligned} \tag{16}$$

where $\boldsymbol{S}(\cdot)$ is the operator performing the cross product between two (3×1) vectors. Given $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$, $\boldsymbol{S}(\boldsymbol{\omega})$ takes on the form

$$\boldsymbol{S}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \tag{17}$$

The angular velocities of Σ_{CR} and Σ_M with respect to Σ_W are represented by ${}^W\boldsymbol{\omega}_{CR}$ and ${}^W\boldsymbol{\omega}_M$, and the angular velocity of Σ_M with respect to Σ_{CR} is represented by ${}^{CR}\boldsymbol{\omega}_M$. Then the following relations hold:

$$ {}^{CR}\boldsymbol{\omega}_M = {}^{CR}\boldsymbol{R}_W(\boldsymbol{q})({}^W\boldsymbol{\omega}_M - {}^W\boldsymbol{\omega}_{CR}). \tag{18} $$

The target's 3-D pose volecity is defined as

$$ {}^{CR}\dot{\boldsymbol{\psi}}_M = \begin{bmatrix} {}^{CR}\dot{\boldsymbol{r}}_M \\ {}^{CR}\dot{\boldsymbol{\epsilon}}_M \end{bmatrix}, \tag{19} $$

where the time derivation of target's position ${}^{CR}\dot{\boldsymbol{r}}_M$ is given by (16). The relation between the time derivative of ${}^{CR}\boldsymbol{\epsilon}_M$ and the body angular velocity ${}^{CR}\boldsymbol{\omega}_M$ is given by [14] and is rewrote as

$$ {}^{CR}\dot{\boldsymbol{\epsilon}}_M = \frac{1}{2}({}^{CR}\eta_M\boldsymbol{I} - \boldsymbol{S}({}^{CR}\boldsymbol{\epsilon}_M)){}^{CR}\boldsymbol{\omega}_M, \tag{20} $$

where ${}^{CR}\boldsymbol{\omega}_M$ is given by (18).

Moreover, the camera velocity, which is considered as the end-effector velocity, can be expressed using the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q}) = [\boldsymbol{J}_P{}^T(\boldsymbol{q}), \boldsymbol{J}_O{}^T(\boldsymbol{q})]^T$,

$$ {}^W\dot{\boldsymbol{r}}_{CR} = \boldsymbol{J}_P(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{21} $$

$$ {}^W\boldsymbol{\omega}_{CR} = \boldsymbol{J}_O(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{22} $$

$$ \begin{aligned} \boldsymbol{S}({}^{CR}\boldsymbol{\omega}_W) &= -{}^{CR}\boldsymbol{R}_W(\boldsymbol{q})\boldsymbol{S}({}^W\boldsymbol{\omega}_{CR}){}^W\boldsymbol{R}_{CR}(\boldsymbol{q}) \\ &= -{}^{CR}\boldsymbol{R}_W(\boldsymbol{q})\boldsymbol{S}(\boldsymbol{J}_O(\boldsymbol{q})\dot{\boldsymbol{q}}){}^W\boldsymbol{R}_{CR}(\boldsymbol{q}). \end{aligned} \tag{23} $$

Substituting (21), (22), (23) to (16), (20), the target velocity in ^{CR} ${}^{CR}\dot{\boldsymbol{\psi}}_M$ can be described by a mathematical formulation:

$$ {}^{CR}\dot{\boldsymbol{\psi}}_M = \begin{bmatrix} {}^{CR}\dot{\boldsymbol{r}}_M \\ {}^{CR}\dot{\boldsymbol{\epsilon}}_M \end{bmatrix} $$

$$ = \begin{bmatrix} -{}^{CR}\boldsymbol{R}_W(\boldsymbol{q})\boldsymbol{J}_P(\boldsymbol{q}) + {}^{CR}\boldsymbol{R}_W(\boldsymbol{q}) \\ \boldsymbol{S}({}^W\boldsymbol{R}_{CR}(\boldsymbol{q}){}^{CR}\boldsymbol{r}_M)\boldsymbol{J}_O(\boldsymbol{q}) \\ -\frac{1}{2}[{}^{CR}\eta_M\boldsymbol{I} - \boldsymbol{S}({}^{CR}\boldsymbol{\epsilon}_M)]{}^{CR}\boldsymbol{R}_W(\boldsymbol{q})\boldsymbol{J}_O(\boldsymbol{q}) \end{bmatrix} \dot{\boldsymbol{q}} + $$

$$ \begin{bmatrix} {}^{CR}\boldsymbol{R}_W(\boldsymbol{q}) & 0 \\ 0 & \frac{1}{2}[{}^{CR}\eta_M\boldsymbol{I} - \boldsymbol{S}({}^{CR}\boldsymbol{\epsilon}_M)]{}^{CR}\boldsymbol{R}_W(\boldsymbol{q}) \end{bmatrix} \begin{bmatrix} {}^W\dot{\boldsymbol{r}}_M \\ {}^W\boldsymbol{\omega}_M \end{bmatrix} $$

$$ = \boldsymbol{J}_M(\boldsymbol{q}, {}^{CR}\boldsymbol{\psi}_M)\dot{\boldsymbol{q}} + \boldsymbol{J}_N(\boldsymbol{q}){}^W\dot{\boldsymbol{\phi}}_M \tag{24} $$

The matrix $\boldsymbol{J}_M$ in (24) describes how target pose change in Σ_{CR} with respect to changing manipulator pose in Σ_{CR}. The matrix $\boldsymbol{J}_N$ in (24) describes how target pose change in Σ_{CR} with respect to the pose changing of itself in real word.

In this paper, we do not deal with the prediction of the target's motion in the real world, we take account of the prediction of the target velocity in Σ_{CR} based on the joint velocity $\dot{\boldsymbol{q}}$ of the manipulator, so we can rewrite (24) as

$$ {}^{CR}\dot{\boldsymbol{\psi}}_M = \boldsymbol{J}_M(\boldsymbol{q}, {}^{CR}\boldsymbol{\psi}_M)\dot{\boldsymbol{q}}. \tag{25} $$

Then the 3-D pose of the target at time $t + \Delta t$ can be predicted based on the motion of the end-effector motion at time t, presented by

$$ {}^{CR}\hat{\boldsymbol{\psi}}_M(t + \Delta t) = {}^{CR}\boldsymbol{\psi}_M(t) + {}^{CR}\dot{\boldsymbol{\psi}}_M\Delta t. \tag{26} $$

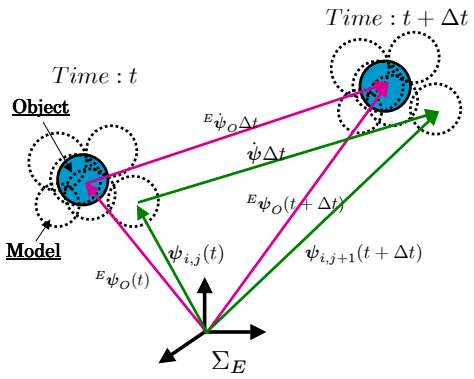


Fig. 3. MFF Compensation. Notice Σ_E and Σ_O are relative coordinates, here we suppose the end-effector is moving and the target is static.

B. MFF Compensation

In the same way as the above equation (25), (26), the pose of the individuals $\boldsymbol{\psi}_{i,j}$ in the next $j+1$ generation can be predicted based on the current pose,

$$ \dot{\boldsymbol{\psi}} = \boldsymbol{J}_M(\boldsymbol{q}, \hat{\boldsymbol{\psi}}(t))\dot{\boldsymbol{q}}. \tag{27} $$

$$ \boldsymbol{\psi}_{i,j+1}(t + \Delta t) = \boldsymbol{\psi}_{i,j}(t) + \dot{\boldsymbol{\psi}}\Delta t, \tag{28} $$

where Δt is the time cost in one generation. By using (28), GA group will move together with the motion of the target in Σ_E, never lose it even under a high-speed moving of robot manipulator, as shown in Fig. 3. Since the effect on the recognition from the dynamics of manipulator can be compensated, recognition by hand-eye cameras will be independent of the dynamics of the manipulator, robust recognition can be obtained just like using fixed cameras.

IV. Controller

A. Desired-trajectory generation

The desired relative relationship of Σ_M and Σ_{CR} is given by Homogeneous Transformation as ${}^{CRd}\boldsymbol{T}_M(t)$, the difference of the desired camera pose Σ_{CRd} and the actual camera pose Σ_{CR} is denoted as ${}^{CR}\boldsymbol{T}_{CRd}$. ${}^{CR}\boldsymbol{T}_{CRd}$ can be described by

$$ {}^{CR}\hat{\boldsymbol{T}}_{CRd}(t) = {}^{CR}\hat{\boldsymbol{T}}_M(t)\ {}^{CRd}\boldsymbol{T}_M^{-1}(t), \tag{29} $$

Notice that (29) is a general deduction that satisfies arbitrary object motion ${}^W\boldsymbol{T}_M(t)$ and arbitrary objective of visual servoing ${}^{CRd}\boldsymbol{T}_M(t)$.

Differentiating (29) with respect to time yields

$$ {}^{CR}\dot{\boldsymbol{T}}_{CRd}(t) = {}^{CR}\dot{\hat{\boldsymbol{T}}}_M(t){}^{CRd}\boldsymbol{T}_M^{-1}(t) + {}^{CR}\hat{\boldsymbol{T}}_M(t){}^{CRd}\dot{\boldsymbol{T}}_M^{-1}(t). \tag{30} $$

Here, ${}^{CRd}\boldsymbol{T}_M(t)$, ${}^{CRd}\dot{\boldsymbol{T}}_M(t)$ are given as the desired visual servoing objective. ${}^{CR}\hat{\boldsymbol{T}}_M(t)$ is mearsured by cameras using the on-line recognition method proposed in Section II. ${}^{CR}\dot{\hat{\boldsymbol{T}}}_M(t)$ is then calculated by

$$ {}^{CR}\dot{\hat{\boldsymbol{T}}}_M(t) = ({}^{CR}\hat{\boldsymbol{T}}_M(t) - {}^{CR}\hat{\boldsymbol{T}}_M(t - \Delta t))/\Delta t, \tag{31} $$

which is output periodically with a time of Δt regardless the object is moving or not. Notice that (25) can not be used to calculate ${}^{CR}\dot{\hat{\boldsymbol{T}}}_M(t)$ because it assumes the object is stationary.

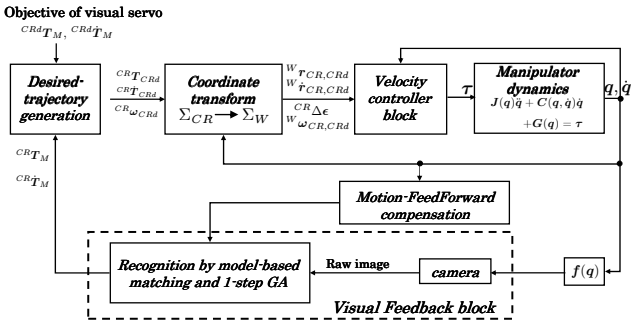


Fig. 4. block diagram of the visual servoing system

B. Servoing controller

The aforementioned real-time recognition system is depicted at the lower side of the block diagram of the visual servoing system in Fig.4. Based on the above analysis of the desired-trajectory generation, the desired hand velocity ${}^{W}\dot{\boldsymbol{r}}_d$ is calculated as,

$$ {}^{W}\dot{\boldsymbol{r}}_d = \boldsymbol{K}_{P_p}\,{}^{W}\boldsymbol{r}_{CR,CRd} + \boldsymbol{K}_{V_p}\,{}^{W}\dot{\boldsymbol{r}}_{CR,CRd}, \tag{32} $$

where ${}^{W}\boldsymbol{r}_{CR,CRd}$, ${}^{W}\dot{\boldsymbol{r}}_{CR,CRd}$ are given by transforming ${}^{CR}\boldsymbol{T}_{CRd}$ and ${}^{CR}\dot{\boldsymbol{T}}_{CRd}$ from Σ_{CR} to Σ_W. $\boldsymbol{K}_{P_p}$ and $\boldsymbol{K}_{V_p}$ are positive definite matrix to determine PD gain.

The desired hand angular velocity ${}^{W}\boldsymbol{\omega}_d$ is calculated as,

$$ {}^{W}\boldsymbol{\omega}_d = \boldsymbol{K}_{P_o}\,{}^{W}\boldsymbol{R}_{CR}\,{}^{CR}\Delta\boldsymbol{\epsilon} + \boldsymbol{K}_{V_o}\,{}^{W}\boldsymbol{\omega}_{CR,CRd}, \tag{33} $$

where ${}^{CR}\Delta\boldsymbol{\epsilon}$ is the quaternion error that from the recognition result directly, and ${}^{W}\boldsymbol{\omega}_{CR,CRd}$ can be calculated by transforming ${}^{CR}\boldsymbol{T}_{CRd}$ and ${}^{CR}\dot{\boldsymbol{T}}_{CRd}$ from Σ_{CR} to Σ_W. Also, $\boldsymbol{K}_{P_o}$ and $\boldsymbol{K}_{V_o}$ are suitable feedback matrix gains.

The desired joint variable $\dot{\boldsymbol{q}}_d$ is obtained by

$$ \dot{\boldsymbol{q}}_d = \boldsymbol{J}^{+}(\boldsymbol{q}) \begin{bmatrix} {}^{W}\dot{\boldsymbol{r}}_d \\ {}^{W}\boldsymbol{\omega}_d \end{bmatrix}. \tag{34} $$

where $\boldsymbol{J}^{+}(\boldsymbol{q})$ is the pseudoinverse matrix of $\boldsymbol{J}(\boldsymbol{q})$, and $\boldsymbol{J}^{+}(\boldsymbol{q}) = \boldsymbol{J}^{T}(\boldsymbol{J}\boldsymbol{J}^{T})^{-1}$. The hardware control system of the velocity-based servo system of PA10 is expressed as

$$ \boldsymbol{\tau} = \boldsymbol{K}_{SP}(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}) + \boldsymbol{K}_{SI}\int_0^t (\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}})dt \tag{35} $$

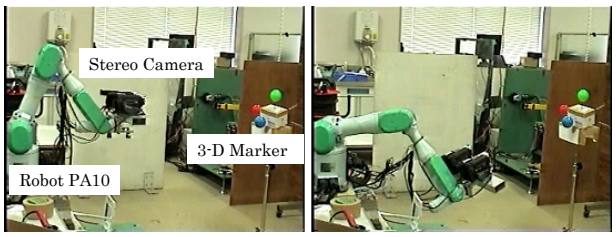
where $\boldsymbol{K}_{SP}$ and $\boldsymbol{K}_{SI}$ are symmetric positive definite matrix to determine PI gain(Table II).

V. Experiment of Visual Servoing

To verify the effectiveness of the proposed visual servoing system, we conduct the experiment of visual servoing to a 3D marker that is composed of a red ball, a green ball and a blue ball. The radiuses of these three balls are set as 30[mm].

TABLE II
GAIN PARAMETERS

Link Number	[L1 L2 L3 L4 L5 L6 L7]
$\boldsymbol{K}_{SP}$	[3200 3200 1400 1400 1000 1000 1000]
$\boldsymbol{K}_{SI}$	[1362 1362 596 596 596 426 426]



(a) (b)

Fig. 5. Step response experiment. (a) Initial pose of Pa10. (b) Visual servoing to a static 3-D marker.

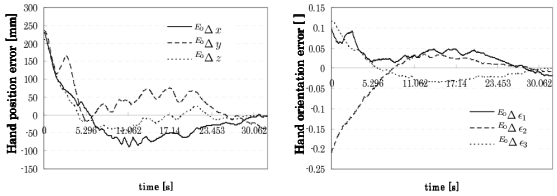


Fig. 6. Hand pose error of step response without using MFF

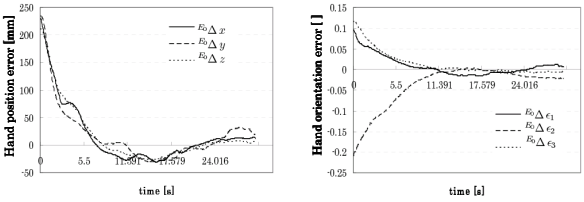


Fig. 7. Hand pose error of step response by using MFF

A. Experimental Condition

A photograph of our experimental system is shown in Fig. 11. The robot used in this experimental system is a 7-Link manipulator, Mitsubishi Heavy Industries PA-10 robot. Two cameras are mounted on the robot manipulator's end-effector. The image processing board, CT-3001, receiving the image from the CCD camera is connected to the DELL Optiplex GX1 (CPU: Pentium2, 400 MHz) host computer.

The initial pose of the end-effector is defined as Σ_{E_0}, and given by

$$ {}^{W}\boldsymbol{T}_{E_0} = \begin{bmatrix} 0 & 0 & 1 & -918 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 455 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{36} $$

position unit: [mm].

B. Experimental Results

1) Step Response Experiment: Here, a static object is set as ${}^{CR}\boldsymbol{\psi}_M = [-70[mm], 70[mm], 1000[mm], 0.1, -0.2, 0.12]^T$, where the value of orientation 0.1 in quaternion expression is about 12[deg]. The objective of visual servoing is given by a fixed relation between Σ_{CR} and Σ_M, as

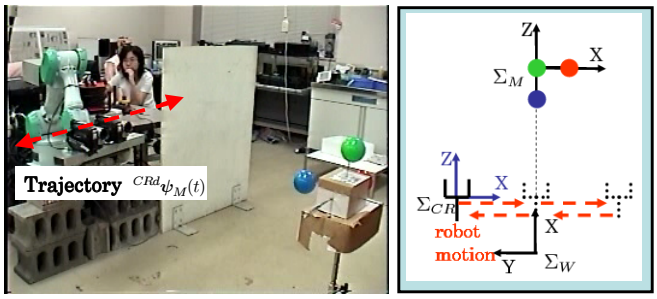
$$ {}^{CR}\boldsymbol{\psi}_{Md} = [0[mm], 10[mm], 900[mm], 0, 0, 0]^T. \tag{37} $$

The initial pose of the robot manipulator is shown in Fig.5(a), and the moved robot manipulator to satisfy ${}^{CR}\boldsymbol{\psi}_{Md}$ is shown in Fig.5(b).

TABLE III

REVIEW OF LITERATURE

Reference	Convergence time of step response
[5]	about $9.9[s]$ when the desired position is parallel to the image plane, else, about $49.5[s]$.
[6]	in x,y, roll,pitch, yaw 30s, in z position about 70s
[15]	about $60s$.
[16]	about 150s.
[17]	is about 200s.



(a) (b)

Fig. 8. (a)A photograph of time-varying visual servo system. (b)Coordinate system of (a).

To show the effectiveness of the proposed MFF method, we perform the step response experiment with MFF method and without MFF method separately. Fig 6 shows the difference of the desired hand pose and the actual hand pose in Σ_{E_0} without using MFF method. Fig 7 shows the hand difference with using MFF method. In Fig 6, the end-effector is unstable from 6[s] to 28[s]. Since the hand began to move, the object in camera frame was moving together with the end-effector, then the recognition dynamics became worse, which cause the vibration in this period. The end-effector cost 30[s] to be converged to the desired pose in the case of not using MFF compensation.

On the other hand, as shown in Fig 7, such vibrations existing in Fig 6 had been suppressed, and the end-effector cost about 10[s] to converge to the desired pose by using MFF method.

Step response is usually used to evaluate the ability of a visual servoing system. Here, we list some similar visual servoing researches and their convergence times in Table III. By comparing the convergence speed with these researches, our system shows a good ability in visual servoing task.

2) Time-varying Path Control Experiment: The visual servoing described in this section is that the object remains stationary and the robot is commanded to move through a reference path with respect to it. Such a visual servoing has been performed by William J. Wilson etc. in [4], and they named it as relative path control experiment.

Here, a static object is set as ${}^{CR}\boldsymbol{\psi}_M = [0[mm], 70[mm], 1300[mm], 0, 0, 0]^T$. The desired end-effector's time-varying trajectory is given by

$$\begin{cases} {}^{CRd}x_M(t) = 100sin(\frac{2\pi}{T}t)[mm] \\ {}^{CRd}y_M(t) = 70[mm] \\ {}^{CRd}z_M(t) = 1300[mm] \\ {}^{CRd}\epsilon_{1M}(t) = 0 \\ {}^{CRd}\epsilon_{2M}(t) = 0 \\ {}^{CRd}\epsilon_{3M}(t) = 0 \end{cases} \quad (38)$$

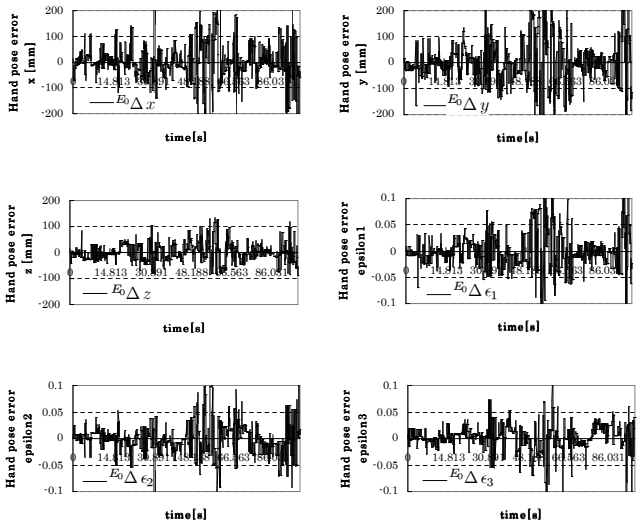


Fig. 9. Hand pose error of time-varying visual servoing without MFF method when $t = 60s$

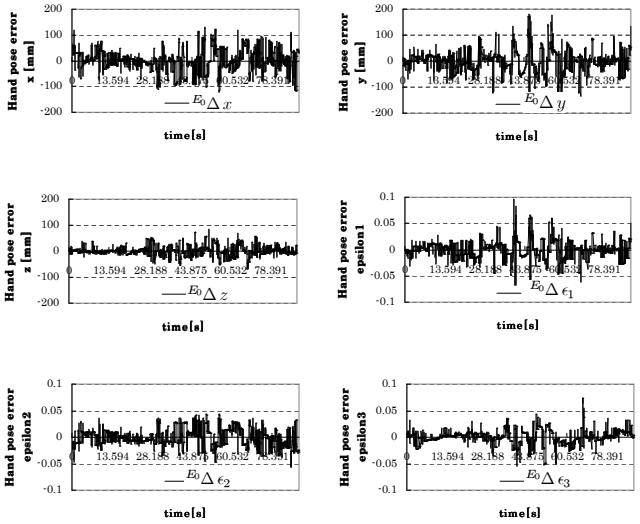


Fig. 10. Hand pose error of time-varying visual servoing with MFF method when $t = 60s$

The desired motion of the end-effector with respect to a static object is shown in Fig. 8.

First, we set the motion period of the manipulator T as 60[s]. Fig 9 shows the errors of the desired hand pose and the actual hand pose in Σ_{E_0} without using MFF method. Fig 10 shows the hand errors with using MFF method. The stereo cameras were shaking because of the dynamics of the robot manipulator. Thus, the fictional motion of the target object coming from the moving camera was difficult to recognize. As shown in Fig.2, the incorrect recognition affects the hand motion directly, and will cause the feedback system unstable. The increased errors shown in Fig 9 indicated the system became unstable as time passing. Compared with Fig.9, the errors in Fig 10 were limited in a range, which means the system was under a stable control.

We compare the visual servoing with and without MFF

TABLE IV

RESULTS OF TIME-VARYING VISUAL SERVOING

Condition	$\bar{F}$	${}^{E_0}\Delta\tilde{x}[mm]$	${}^{E_0}\Delta\tilde{y}[mm]$	${}^{E_0}\Delta\tilde{z}[mm]$	${}^{E_0}\Delta\tilde{\epsilon_1}$	${}^{E_0}\Delta\tilde{\epsilon_2}$	${}^{E_0}\Delta\tilde{\epsilon_3}$
$T = 60s$, without MFF.	0.8416	73.92	92.06	37.25	0.035	0.029	0.025
$T = 60s$, with MFF.	0.9032	51.00	52.85	21.46	0.020	0.020	0.014
$T = 40s$, without MFF.	0.7822	80.39	82.94	37.92	0.032	0.041	0.034
$T = 40s$, with MFF.	0.9052	43.78	51.48	18.16	0.019	0.018	0.014
$T = 20s$, without MFF.	0.7241	96.39	76.48	31.63	0.022	0.045	0.043
$T = 20s$, with MFF.	0.9068	51.04	56.46	21.43	0.022	0.019	0.015

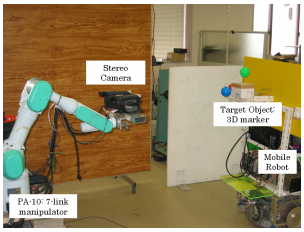


Fig. 11. A photograph of visual servo system

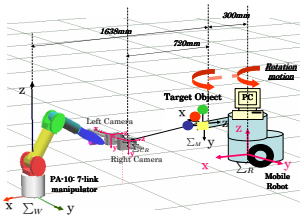


Fig. 12. Coordinate system of visual servoing

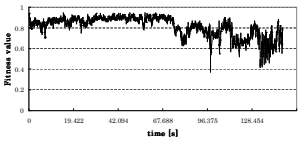


Fig. 15. Fitness value of visual servoing without MFF method

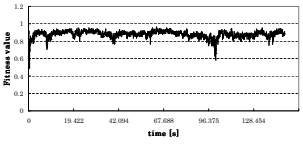


Fig. 16. Fitness value of visual servoing with MFF method

method by changing the servoing speed of end-effector, that is, changing the period as $T = 60, 40, 20[s]$. The figures for comparisons under $T = 40[s]$ and $T = 20[s]$ are not shown here for brevity. Table IV shows the mean value of the fitness function F defined as $\bar{F}$ and the rms (root-mean-square) value of all components of ${}^{E_0}\tilde{\boldsymbol{\psi}}$ defined as ${}^{E_0}\Delta\tilde{\boldsymbol{\psi}} = [{}^{E_0}\Delta\tilde{x}, {}^{E_0}\Delta\tilde{y}, {}^{E_0}\Delta\tilde{z}, {}^{E_0}\Delta\tilde{\epsilon_1}, {}^{E_0}\Delta\tilde{\epsilon_2}, {}^{E_0}\Delta\tilde{\epsilon_3}]^T$ in each situation of $T = 60, 40, 20[s]$. We can see that without using MFF, $\bar{F}$ gets lower, and ${}^{E_0}\Delta\tilde{\boldsymbol{\psi}}$ gets bigger to about 96[mm] in $\Delta\tilde{x}_M$, and 0.045 in $\Delta\tilde{\epsilon_2}_M$ (corresponding to 7[deg]) when T is 20[s], which means the motion of the end-effector became more unstable. On the other hand, by using MFF, both $\bar{F}$ and ${}^{E_0}\Delta\tilde{\boldsymbol{\psi}}$ are not changed much, about 51[mm] in $\Delta\tilde{z}_M$ and 0.019 in $\Delta\tilde{\epsilon_2}_M$ (corresponding to 2[deg]), which indicates the motion of the end-effector kept stable, even the hand-eye cameras move faster and faster.

This time-varying path control experiment has confirmed the effectiveness of the proposed MFF method. By using MFF method, the affect on recognition from the motion of the camera itself is compensated and the recognition dynamics is improved, therefore, the stability of the visual servoing system is increased.

3) Visual Servoing To A Moving Object: In this experiment, the target object is fixed on a mobile robot, as shown in Fig. 11. Fig. 12 shows the coordinate system corresponding to Fig. 11. The coordinate system of the mobile robot is represented as Σ_R. Here, the motion of the mobile robot is rotation around the z axis of Σ_R by

$$\theta_d[deg] = asin(\frac{2\pi}{T})t, \tag{39}$$

where we set $a = 8[deg], T = 40[s]$. The voltage of the right and left wheel is given by

$$V_R = kp(\theta_d - \theta) + kv(\dot{\theta}_d - \dot{\theta}), \tag{40}$$

$$V_L = -V_R, \tag{41}$$

where kp and kv are suitable feedback PD control gains.

The effectiveness of the proposed visual servoing are evaluated by comparing the actual hand pose with the desired hand pose through visual servoing to the moving target object. We also do the same experiment in the case of without using MFF method and with MFF method separately. Here, the objective of visual servoing is a fixed relative pose between Σ_M and Σ_{CR}, defined as ${}^{CRd}\boldsymbol{\psi}_M = [0, 10, 700, 0, 0, 0]^T$.

Figs. 13(a) to (f) is the experimental results in the case of not using MFF method, which show the actual motion of the end-effector with respect to the fixed frame of Σ_{E_0}, defined as ${}^{E0}\boldsymbol{\psi}_E$, compared with the desired hand pose ${}^{E0}\boldsymbol{\psi}_{Ed}$. Figs. 14(a) to (f) show the experimental results of ${}^{E0}\boldsymbol{\psi}_E$ and ${}^{E0}\boldsymbol{\psi}_{Ed}$ in the case of using MFF method. In the period of the trajectory of ${}^{E0}\boldsymbol{\psi}_{Ed}$ is a straight line, the mobile robot did not move, visual servoing to a static object was performed firstly. Then the desired trajectory in Fig. 13 and Fig. 14(a),(e) began to turn to curved line of sin/cos function, the mobile robot started to move. Comparing Figs. 13(a),(e) with Fig. 14(a),(e), the time-delay of hand motion in the case of using MFF method was smaller than that without using MFF method. The errors of hand motion in the other (b),(c),(d),(f) figures were also smaller in the case of using MFF method.

The Fitness values representing the accuracy of the object recognition during visual servoing are shown in Fig. 15 and Fig. 16, corresponding Figs. 13 and Figs. 14 in the case of without using MFF and with MFF separately. It can be found that the recognition accuracy of the object decreased as soon as the start of the hand motion without using MFF compensation, as the fitness values shown in Fig. 15 became lower from 65[s]. The undesired influence of the recognition and the hand motion in the feedback loop of the visual servoing system discussed in the Introduction has been shown in Fig. 15 and Figs. 13. However, by using MFF method, such an undesired circulation can be cut down, and both the recognition and the hand motion are stable, which has been verified in Fig. 16 and Figs. 14.

VI. CONCLUSION

This paper deals with position-based 6-DoF visual servoing. We propose a MFF method to compensate the fic-

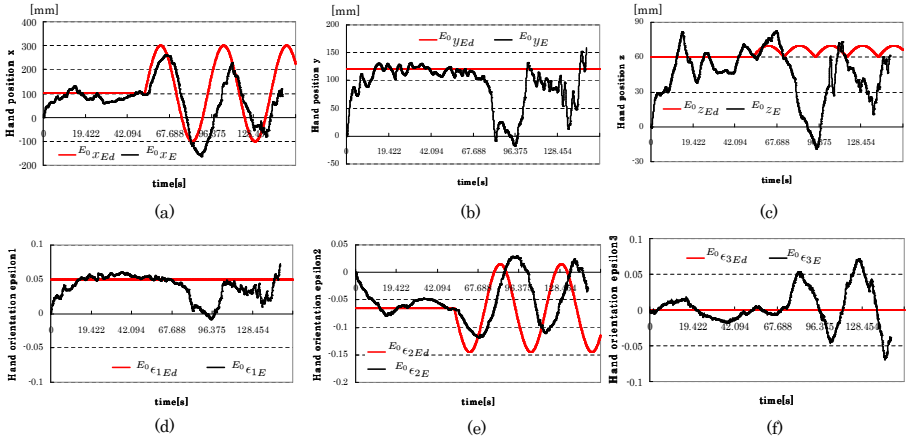


Fig. 13. Hand pose error of visual servoing without MFF. method

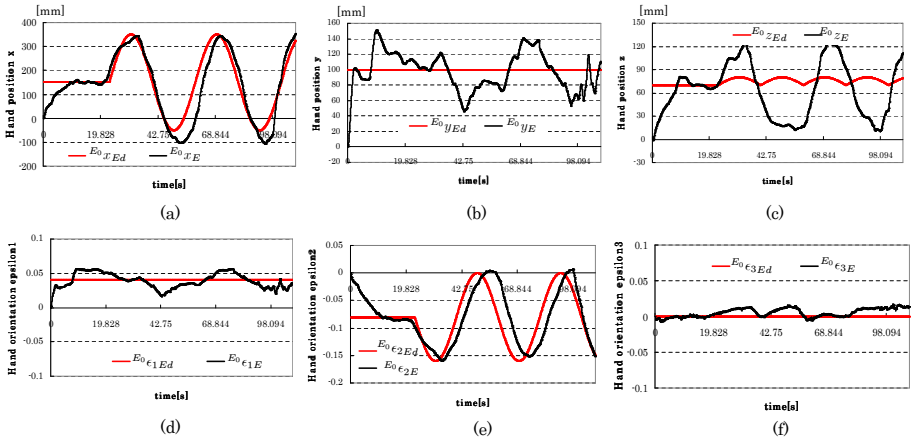


Fig. 14. Hand pose error of visual servoing with MFF. method

tional motion of the target based on the joint velocity of manipulator, and extract the real motion of the target for the robot to recognize during visual servoing. Thus, visual recognition preciseness is improved, and the visual servoing become more stable. Three experiments has been conducted, and the effectiveness of our proposed visual servoing system has been confirmed.

REFERENCES

[1] V.Lippiello, B.Siciliano and L.Villani "Position-Based Visual Servoing in Industrial Multirobot Cells Using a Hybrid Camera Configuration", IEEE Trans on Robotics Vol. 23, Issue 1, 2007 Page(s):73 - 86

[2] S.Hutchinson, G.Hager, and P.Corke, "A Tutorial on Visual Servo Control", IEEE Trans. on Robotics and Automation, vol. 12, no. 5, pp. 651-670, 1996.

[3] Wolfgang Sepp, Stefan Fuchs and Gerd Hirzinger, "Hierarchical Featureless Tracking for Position-Based 6-DoF Visual Servoing", IROS2006.

[4] William J. Wilson, Carol C. Williams Hulls and Graham S. Bell, "Relative End-effector Control Using CartesianPosition Based Visual Servoing", IEEE Trans. on Robotics and Automation, vol. 12, No. 5, 1996, pp. 684-696.

[5] Omar Tahri and Francois Chaumette, "Point-Based and Region-Based Image Moments for Visual Servoing of Planar Objects", IEEE Tran. on Robotics, vol. 21, no. 6, Dec 2005.

[6] Tarek Hamel and Robert Mahony, "Visual Servoing of an Under-Actuated Dynamic Rigid-Body System: An Image-Based Approach", IEEE Trans. on Robotics and Automation, VOL. 18, NO. 2, APRIL 2002.

[7] H. Suzuki, M. Minami, "Visual Servoing to catch fish Using Global/local GA Search", IEEE/ASME Transactions on Mechatronics, Vol.10, Issue 3, 352-357 (2005.6).

[8] R.Yang and Z.Zhang, "Model-based head pose tracking with stereovision", Proc.5th IEEE Int.Conf.on Automatic Face and Gesture Recogniton, 2002.

[9] L.Masson, M. Dhome and F.Jurie "Robust Real Time Tracking of 3D Objects" Proc.17th IEEE Int.Conf.on Patten Recognition, 2004.

[10] Y.Maeda, G.Xu, "Smooth Matching of Feature and Recovery of Epipolar Equation by Tabu Search", IEICE, Vol.J83-D-2, No.3, pp.440-448, 1999.

[11] S.Yamane, M.Izumi, K.Fukunaga, "A Method of Model-Based Pose Estimation", IEICE, Vol.J79-D-2, No.2, pp.165-173, Feb, 1996.

[12] F.Toyama, K.Shoji, J.Miyamichi, "Pose Estimation from a Line Drawing Using Genetic Algorithm", IEICE, Vol.J81-D-2, No.7, pp.1584-1590, July, 1998.

[13] W. Song, M. Minami, S. Aoyagi, "On-line Stable Evolutionary Recognition Based on Unit Quaternion Representation by Motion-Feedforward Compensation", International Journal of Intelligent Computing in Medical Sciences and Image Processing (IC-MED) Vol. 2, No. 2, Page 127-139 (2007).

[14] B.Siciliano and L.Villani: *Robot Force Control*, ISBN 0-7923-7733-8.

[15] E.Malis, F.Chaumentte and S.Boudet, "2-1/2-D Visual Servoing", IEEE Trans. on Robotics and Automation, vol. 15, no. 2, pp. 238-250, 1999.

[16] Peter I. Corke and Seth A. Hutchinson, "A New Partitioned Approach to Image-Based Visual Servo Control", IEEE Trans. on Robotics and Automation, VOL. 17, NO. 4, August 2001.

[17] Nicholas R. Gans, and Seth A. Hutchinson, "Stable Visual Servoing Through Hybrid Switched-System Control", IEEE Tran. on Robotics, vol. 23, no. 3, JUNE 2007.