

3D Visual Servoing by Feedforward Evolutionary Recognition*

Wei SONG**, Yu FUJIA*** and Mamoru MINAMI***

** School of Mechatronics Engineering and Automation, Shanghai University
149 Yanchang Road, Zhabei District, Shanghai, 200072, China

*** Graduate School of Natural Science and Technology, Okayama University
1-1-1 Tsushima-naka, Kita-ku, Okayama-shi, 700-8530, Japan
E-mail: minami@qumi.sys.okayama-u.ac.jp

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 feedforward (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.

Key words : Evolutionary Recognition, Visual Servoing, Hand-eye Motion Feedforward

1. 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 analysis is well-known in a feedback control theory. Let dY denote the change of the output Y , it gives

$$\frac{dY}{Y} = \frac{1}{1 + CSH} \frac{dS}{S}. \quad (1)$$

Usually $CSH \gg 1$, 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

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

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

$$\frac{dY}{Y} \approx -\frac{dH}{H}. \quad (3)$$

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 affect 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

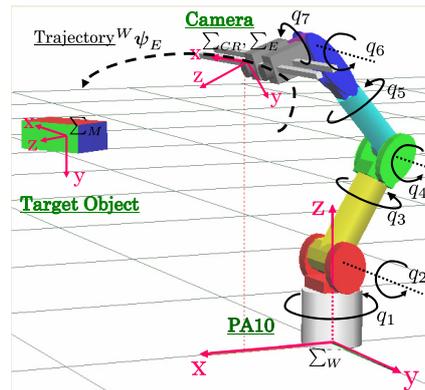


Fig. 1 Visual servo system of PA-10

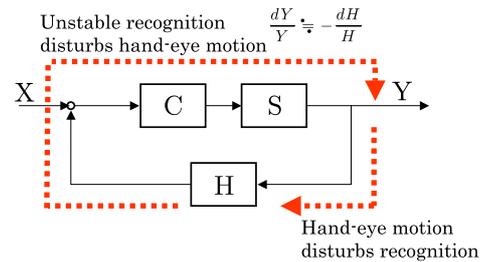


Fig. 2 Feedback system

error. This repeating 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 this paper, we pay attention to a visual servoing system with a hand-eye configuration, having the camera mounted on the robot's end-effector. In this case, the dynamics of the manipulator will make the recognition unit deteriorating directly, since the manipulator oscillation produces a false motion of the target object in the camera image even though the target is stopping in the task space. We call the false motion as "fictional motion". A motion-feedforward (MFF) method is used to improve the deterioration in recognition dynamics caused by the fictional motion. The target's 3-D pose in the camera image made by the fictional motion can be predicted by using the kinematics of the manipulator based on the current estimated pose⁽¹⁾⁽²⁾. However, the prediction result of MFF includes errors from the estimation result, so we could not use MFF independently in the visual servoing control system. Here, our proposal is to combine the evolutionary recognition method using GA⁽³⁾ and the MFF, that is "1-step GA + MFF". We use the prediction result to reset the individual pose in the next generation of GA. Thus, the on-line optimization method, "1-Step GA" combined with the above prediction by MFF has an ability to prevent the pose tracking error from increasing by correcting the pose estimation through exploring nature of GA in heuristic searching behavior. Since the fictional motions can be compensated during on-line estimation in GA process, it seems that the recognition was performed by using just fixed cameras in task space, then the recognition dynamics can be separated from the dynamics of the manipulator. Thus the tracking becomes easier and the tracking dynamics can be improved.

Visual servoing can be classified into two major groups: position-based and image-based visual servoing⁽⁴⁾. The advantages and drawbacks of each visual servoing method have been discussed by a significant amount of researches⁽⁵⁾⁽⁶⁾⁽⁷⁾. Compared with image-based visual servoing, position-based visual servoing is more understandable, since the way of the visual servo is more like human-being, that is, to determine the object pose in Cartesian coordinate frame and lead to Cartesian robot motion planning. Moreover, in position-based visual servoing, the robot controller and object pose recognition are separated as independent unit. So MFF compensation suits to be applied in position-based visual servoing, since the robot motion is feedforwarded to the recognition unit to compensate the fictional motion caused by the hand-eye camera's dynamical motion.

There are some researches on position-based visual servoing that used extended kalman filters to predict the target's pose in the real world⁽⁵⁾⁽⁸⁾. They considered from the view point of the object real motion, but did not pay attention to the fictional motion caused by the hand-eye camera's dynamical motion. 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. The fictional motions is compensated during on-line recognition by MFF, then the recognition dynamics can be improved and stable visual servoing can be achieved.

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 compared 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. Time-varying path control visual servoing was firstly proposed by William J. Wilson etc.⁽⁵⁾, in which the target object is static, and a trajectory with respect to the object is given to direct the manipulator. After⁽⁵⁾, few researches discussed about this kind of visual servoing. Since the object is static, only fictional motions caused by the hand-eye camera's dynamical motions exist in the system, so our proposed on-line recognition by MFF method is specifically effective in this case that we will show in this paper. 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.

2. On-line Evolutionary Recognition

2.1. Background of 3-D Object Recognition

There is a variety of approaches for 3D target object pose estimation, and they can be classified into three general categories: (1) feature-based, (2) appearance-based, and (3) model-based.

Feature-based approaches use local features like points, line segments, edges, or regions to match against the incoming video input to update estimating pose. Feature-based techniques are naturally less sensitive to occlusions than other methods, 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⁽¹¹⁾. Appearance-based approach is based on a technique to minimize sum-of-squared difference calculated by reference image and input image, e.g., Gauss-Newton method⁽¹²⁾, Varying Gauss-Newton method⁽¹³⁾, Second-order Minimization method⁽¹⁴⁾⁽¹⁵⁾, which can avoid to calculate Hessian matrix while keeping high convergence rate. These methods hereditarily utilize linearization of the sum-of-squared difference to be optimized, confining the tracking ability to be locally valid⁽¹⁶⁾, restricted by optimizing assumption that the local area include only single peak, which may hinder fast motion tracking. Furthermore these methods lack discussions concerning the convergence in time domain, where the tracking problem should be treated as time varying optimization since the target object to be tracked may move in an operational space. Model-based approach 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⁽¹⁷⁾⁽¹⁸⁾. Our method is included in this category.

2.2. 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_E , its position and orientation are determined by six parameters, $\psi = [\mathbf{r}^T, \boldsymbol{\epsilon}^T]^T$, where $\mathbf{r} = [x, y, z]^T$, $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \epsilon_3]^T$. Here, the target's orientation is represented by unit quaternion⁽²⁰⁾, which has an advantage that can represent the orientation of a rigid body without singularities. The unit quaternion, viz. Euler parameters, defined as

$$Q = \{\eta, \boldsymbol{\epsilon}\}, \quad (4)$$

where

$$\eta = \cos \frac{\theta}{2}, \quad \epsilon = \sin \frac{\theta}{2} \mathbf{k}, \quad (5)$$

here, $\mathbf{k}(\|\mathbf{k}\| = 1)$ 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

$$\eta^2 + \epsilon^T \epsilon = 1. \quad (6)$$

In (6) η can be calculated by ϵ , so we just use three parameters ϵ 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 $F(\psi)$ by using the color information of the target. Please refer to⁽¹⁹⁾ for a detailed definition of $F(\psi)$. When the searching models fit to the target objects being imaged in the right and left images, $F(\psi)$ gives the maximum value. Therefore the 3-D object's position/orientation measurement problem can be converted to a searching problem of ψ that maximizes $F(\psi)$. We solve this optimization problem by 1-step GA method that will be explained in the next section.

2.3. GA-based On-line Recognition "1-step GA"

Theoretically optimal pose $\psi_{max}(t)$ that gives the highest peak of $F(\psi(t))$ is defined as

$$\psi_{max}(t) = \left\{ \psi(t) \mid \max_{\psi \in \mathbf{L}} F(\psi(t)) \right\}, \quad (7)$$

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

An individual of GA is defined as $\psi_i^j(t)$, which means the i -th gene ($i = 1, 2, \dots, p$) in the j -th generation, to search $\psi_{max}(t)$. The genes of GA individual representing $\psi_i^j(t)$ is defined by binary strings, which are generated randomly in the initial population, with a given individual number p . Denote $\psi_{max}^{GA}(t)$ to be the maximum among the p genes of $\psi_i^j(t)$ in GA process,

$$\psi_{max}^{GA}(t) = \left\{ \psi_i^j(t) \mid \max_{\psi_i^j \in \mathbf{L}} F(\psi_i^j(t)) \right\}. \quad (8)$$

In fact we cannot always guarantee the best individual of GA $\psi_{max}^{GA}(t)$ should coincide with the theoretically optimal pose $\psi_{max}(t)$, because the number of GA's individuals is not infinite. The difference between $\psi_{max}(t)$ and $\psi_{max}^{GA}(t)$ is denoted as

$$\delta\psi(t) = \psi_{max}(t) - \psi_{max}^{GA}(t). \quad (9)$$

And the difference between $F(\psi_{max}(t))$ and $F(\psi_{max}^{GA}(t))$ is denoted as

$$\Delta F(\delta\psi(t)) = F(\psi_{max}(t)) - F(\psi_{max}^{GA}(t)), \quad (10)$$

Since $F(\psi_{max}(t)) \geq F(\psi_{max}^{GA}(t))$, we have

$$\Delta F(\delta\psi(t)) \geq 0. \quad (11)$$

Based on the definition of $\Delta F(\delta\psi(t))$ in (10), in this research, we let GA's work in the following way:

(a) GA evolves to minimize $\Delta F(\delta\psi(t))$.

(b) The elitist individual of GA is preserved at every generation (elitist gene preservation strategy).

(c) $\psi_{max}^{GA}(t)$ does keep the same value in the evolving when the evolved new gene with different value gives the same value of ΔF .

Here, we present two assumptions.

[Assumption 1] $\Delta F(\delta\psi(t))$ is positive definite.

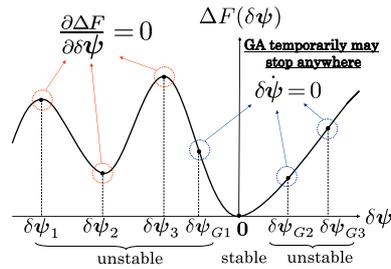


Fig. 3 The invariant set of the solutions of $\Delta\dot{F}(\delta\psi(t)) = 0$.

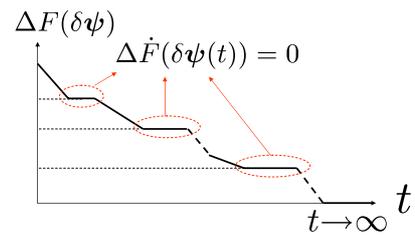


Fig. 4 The changing of $\Delta F(\delta\psi(t))$ with respect to time t in the whole GA's evolution.

This means the distribution of $F(\psi(t))$ satisfies $\Delta F(\delta\psi(t)) = 0$ if and only if $\delta\psi(t) = 0$, which indicates there is a single minimum in the searching space L . ΔF is multipeak distribution having peaks and bottoms with limited number.

[Assumption 2] $\dot{F}(\psi_{max}^{GA}(t)) \geq 0$.

Differentiating (10) by time t , we have

$$\Delta\dot{F}(\delta\psi(t)) = \dot{F}(\psi_{max}(t)) - \dot{F}(\psi_{max}^{GA}(t)). \quad (12)$$

We defined $F(\psi_{max}(t)) = 1$ representing that the true pose of the target object gives the highest peak. Therefore, the time differentiation of $F(\psi_{max}(t))$ will be $\dot{F}(\psi_{max}(t)) = 0$. Thus, from (12) and [Assumption 2], we have

$$\Delta\dot{F}(\delta\psi(t)) = -\dot{F}(\psi_{max}^{GA}(t)) \leq 0. \quad (13)$$

$\psi_{max}^{GA}(t)$ represents current best GA solution. [Assumption 2] means GA can change its best gene $\psi_{max}^{GA}(t)$ to always reduce the value of ΔF regardless of dynamic image or static one, which indicates that the convergence speed to the target in the dynamically continuous images should be faster than the moving speed of the target object.

We cannot guarantee that the above two assumptions always hold, since they depend on some factors such as object's shape, object's speed, definition of $F(\psi(t))$, parameters of GA and viewpoint for observing, lightening environment, et al.. However, we can make efforts to improve the environment and correlation function and so on. Providing above two assumptions be satisfied, (11) and (13) hold, then $\Delta F(\delta\psi(t))$ is so-called Lyapunov function. The objective here is to verify that $\delta\psi(t)$ asymptotically stable, resulting in it converges to $\mathbf{0}$ by using the Lyapunov function of $\Delta F(\delta\psi(t))$, meaning $\psi_{max}^{GA}(t) \rightarrow \psi_{max}(t)$, ($t \rightarrow \infty$), and the following shows how to verify it.

Since $\Delta\dot{F}(\delta\psi(t))$ is only negative semi-definite, in the view of LaSalle theorem, $\delta\psi(t)$ asymptotically converges to the invariant set of the solutions $\delta\psi$ satisfying $\Delta\dot{F}(\delta\psi(t)) = 0$. Considering the following expression,

$$\Delta\dot{F}(\delta\psi(t)) = \frac{\partial\Delta F}{\partial\delta\psi} \cdot \delta\dot{\psi} = 0, \quad (14)$$

the first part $\partial\Delta F/\partial\delta\psi$ describes partial differentiation of ΔF with respect to $\delta\psi$, implying steepest descending direction of ΔF in the space of $\delta\psi$; the second part $\delta\dot{\psi}$ describes the difference between the moving speed of the target object and the evolution speed of the best gene of GA, by the definition in (9).

Equation (14) shows the invariant set of the solutions of $\Delta\dot{F}(\delta\psi(t)) = 0$ includes (1): P_1 , the solution set of $\partial\Delta F/\partial\delta\psi = 0$; (2): P_2 , the solution set of $\delta\dot{\psi} = 0$; and (3): P_3 , the solution set satisfying $\partial\Delta F/\partial\delta\psi \neq \mathbf{0}$, $\delta\dot{\psi} \neq \mathbf{0}$, but their inner product is 0.

As shown in Fig. 3, P_1 includes the points of $\delta\psi$ that give the local maximum or minimum values of the function ΔF including $\mathbf{0}$. The number of these points is finite by [Assumption 1] denoted by p , that is

$$P_1 = \{\mathbf{0}, \delta\psi_1, \delta\psi_2, \dots, \delta\psi_{p-1}\}. \quad (15)$$

The evolving process of GA may stay temporarily at the same ΔF value. If the target object is static, it means the best gene of GA stop at some moments for the reason that the limited individuals of GA could not improve a current solution that gives a smaller fitness function value ΔF during some generations. And when the target object is moving, $\delta\dot{\psi} = 0$ means at these moments that the evolution speed of the best gene of GA is equal to the moving speed of the target object, by (9). The number of these points is assumed to be possibly finite, denoted by q . Thus, we describe the set of P_2 as

$$P_2 = \{\mathbf{0}, \delta\psi_{G1}, \delta\psi_{G2}, \dots, \delta\psi_{G(q-1)}\}. \quad (16)$$

Notice that there is another solution set of $\delta\psi$: P_3 . In this case, the vector of $\partial\Delta F/\partial\delta\psi$ is vertical to the vector of $\delta\dot{\psi}$ since the calculation $(\Delta F/\delta\psi) \cdot \delta\dot{\psi}$ in (14) means inner cross product, which means GA evolves in the direction that keeps a same fitness function value ΔF . This GA's evolution way is forbidden in this research for the GA's work rule (c) that we have stated above. Then, P_3 is null. So the invariant set that $\delta\psi(t)$ asymptotically converges to is

$$P = P_1 \cup P_2. \quad (17)$$

Here, $\delta\psi_1, \delta\psi_2, \dots, \delta\psi_{p-1}$ in P_1 are all unstable, because we define " $F(\psi(t))$ is positive definite and satisfies $\Delta F(\delta\psi(t)) = 0$ if and only if $\delta\psi(t) = \mathbf{0}$ " in [Assumption 1], which means $\Delta F(\delta\psi_i) > 0$ ($i = 1, 2, \dots, p-1$), and only $\delta\psi = \mathbf{0}$ gives minimum value of $\Delta F(\delta\psi(t))$, so only $\delta\psi = \mathbf{0}$ is stable. And in P_2 , all the points are unstable except the point $\mathbf{0}$, for the reason that GA always has possibility to get out of these points by its evolving nature, which has been denoted in the GA's work way (a) that GA evolves to minimize $\Delta F(\delta\psi(t))$.

Therefore, $\mathbf{0}$ is the only stable point in the invariant set of P , that is, $\delta\psi(t)$ will finally converges to $\mathbf{0}$. The image of the changing of $\Delta F(\delta\psi(t))$ with respect to time t in the whole GA's evolution is shown in Fig. 4.

The above verification shows $\delta\psi(t) \rightarrow \mathbf{0}$, which means

$$\psi_{max}^{GA}(t) \rightarrow \psi_{max}(t), \quad (t \rightarrow \infty) \quad (18)$$

Let t_ϵ denotes a convergence time, then

$$|\delta\psi(t)| = |\psi_{max}(t) - \psi_{max}^{GA}(t)| \leq \epsilon, \quad (\epsilon > 0, t \geq t_\epsilon) \quad (19)$$

In (19), ϵ is tolerable extent that can be considered as a observing error. Thus, it is possible to realize real-time optimization, because $\psi_{max}^{GA}(t)$ can be assumed to be in the vicinity of the theoretically optimal $\psi_{max}(t)$ after t_ϵ .

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

$$\psi_i^j(t) \xrightarrow{\text{evolve}} \psi_i^{j+1}(t + \Delta t). \quad (20)$$

Obviously, this time-discrete evolution with the interval of time Δt may enlarge the recognition error $\delta\psi(t)$. Should this undesirable influence of Δt be considered, the tolerable pose error ϵ will expand to ϵ' as,

$$|\delta\psi(t)| \leq \epsilon', \quad (\epsilon' > \epsilon > 0). \quad (21)$$

Since the GA process to recognize the target's pose at the current time is executed only one time with the period of Δt as the current quasi-optimal pose $\psi_{max}^{GA}(t)$ is output synchronously, we named this on-line recognition method as "1-step GA". We have confirmed that the above real-time optimization problem could be solved by "1-step GA" through several experiments to recognize swimming fish⁽³⁾ and human face⁽²¹⁾.

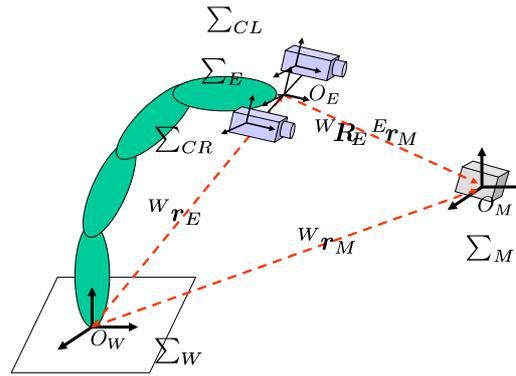


Fig. 5 Coordinate system

3. Hand-eye Motion Compensation

The motion of the target being seen from the hand-eye camera is affected by both the motion of the target in the real world and the ego motion of the hand-eye camera (that is, the motion of the end-effector). Here we describe such a relationship, which can distinguish these two motions in a mathematical formulation.

3.1. Analysis of target's motion in Σ_E

The target coordinate system is represented as Σ_M (see Fig. 5). Take Σ_W as the reference frame. Denote the vector from the origin of Σ_W (O_W) to the origin of Σ_E (O_E) expressed in Σ_W as ${}^W r_E$, the vector from O_W to the origin of Σ_M (O_M) expressed in Σ_W as ${}^W r_M$, and the vector from Σ_E to Σ_M expressed in Σ_E as ${}^E r_M$. The following relations hold:

$${}^E r_M = {}^E R_W(q)({}^W r_M - {}^W r_E(q)), \quad (22)$$

where ${}^E R_W$ is a rotation matrix determined by q . Differentiating (22) with respect to time, we have

$${}^E \dot{r}_M = {}^E R_W(q)({}^W \dot{r}_M - {}^W \dot{r}_E) + S({}^E \omega_W) {}^E R_W(q)({}^W r_M - {}^W r_E(q)). \quad (23)$$

The angular velocities of Σ_E and Σ_M with respect to Σ_W are represented by ${}^W \omega_E$ and ${}^W \omega_M$, and the angular velocity of Σ_M with respect to Σ_E is represented by ${}^E \omega_M$. Then the following relation holds:

$${}^E \omega_M = {}^E R_W(q)({}^W \omega_M - {}^W \omega_E). \quad (24)$$

The 3-D pose of the target expressed in Σ_E is defined as

$${}^E \psi_M = \begin{bmatrix} {}^E r_M \\ {}^E \epsilon_M \end{bmatrix}, \quad (25)$$

Then the velocity of the target's 3-D pose is defined as

$${}^E \dot{\psi}_M = \begin{bmatrix} {}^E \dot{r}_M \\ {}^E \dot{\epsilon}_M \end{bmatrix}, \quad (26)$$

where the time derivation of target's position ${}^E \dot{r}_M$ is given by (23). The relation between the time derivative of ${}^E \epsilon_M$ and the body angular velocity ${}^E \omega_M$ is given by the following equation, (20)

$${}^E \dot{\epsilon}_M = \frac{1}{2}({}^E \eta_M I - S({}^E \epsilon_M)) {}^E \omega_M, \quad (27)$$

where ${}^E \omega_M$ is given by (24).

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

$${}^W \dot{\mathbf{r}}_E = \mathbf{J}_P(\mathbf{q})\dot{\mathbf{q}}, \quad (28)$$

$${}^W \boldsymbol{\omega}_E = \mathbf{J}_O(\mathbf{q})\dot{\mathbf{q}}, \quad (29)$$

$$\mathbf{S}({}^E \boldsymbol{\omega}_W) = -{}^E \mathbf{R}_W(\mathbf{q})\mathbf{S}({}^W \boldsymbol{\omega}_E){}^W \mathbf{R}_E(\mathbf{q}) = -{}^E \mathbf{R}_W(\mathbf{q})\mathbf{S}(\mathbf{J}_O(\mathbf{q})\dot{\mathbf{q}}){}^W \mathbf{R}_E(\mathbf{q}). \quad (30)$$

Substituting (28), (29), (30) to (23), (27), and rewriting ${}^E \boldsymbol{\psi}_M$ as $\boldsymbol{\psi}$, ${}^E \dot{\boldsymbol{\psi}}_M$ as $\dot{\boldsymbol{\psi}}$ for abbreviation, the target velocity $\dot{\boldsymbol{\psi}}_M$ can be described as:

$$\begin{aligned} \dot{\boldsymbol{\psi}} &= \begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\boldsymbol{\epsilon}} \end{bmatrix} \quad (31) \\ &= \begin{bmatrix} -{}^E \mathbf{R}_W(\mathbf{q})\mathbf{J}_P(\mathbf{q}) + {}^E \mathbf{R}_W(\mathbf{q})\mathbf{S}({}^W \mathbf{R}_E(\mathbf{q})\mathbf{r})\mathbf{J}_O(\mathbf{q}) \\ -\frac{1}{2}(\eta \mathbf{I} - \mathbf{S}(\boldsymbol{\epsilon})){}^E \mathbf{R}_W(\mathbf{q})\mathbf{J}_O(\mathbf{q}) \end{bmatrix} \dot{\mathbf{q}} + \begin{bmatrix} {}^E \mathbf{R}_W(\mathbf{q}) & 0 \\ 0 & \frac{1}{2}[\eta \mathbf{I} - \mathbf{S}(\boldsymbol{\epsilon})]{}^E \mathbf{R}_W(\mathbf{q}) \end{bmatrix} \begin{bmatrix} {}^W \dot{\mathbf{r}}_M \\ {}^W \boldsymbol{\omega}_M \end{bmatrix} \\ &= \mathbf{J}_M(\mathbf{q}, \boldsymbol{\psi})\dot{\mathbf{q}} + \mathbf{J}_N(\mathbf{q}, \boldsymbol{\psi}){}^W \dot{\boldsymbol{\phi}}_M \\ &= \begin{bmatrix} \mathbf{J}_M(\mathbf{q}, \boldsymbol{\psi}) & \mathbf{J}_N(\mathbf{q}, \boldsymbol{\psi}) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ {}^W \dot{\boldsymbol{\phi}}_M \end{bmatrix} \\ &= \mathbf{J}_A(\mathbf{q}, \boldsymbol{\psi}) \begin{bmatrix} \dot{\mathbf{q}} \\ {}^W \dot{\boldsymbol{\phi}}_M \end{bmatrix} \quad (32) \end{aligned}$$

The matrix \mathbf{J}_M in (32) describes how target pose change in camera coordinate with respect to the joint velocity of the manipulator $\dot{\mathbf{q}}$. The matrix \mathbf{J}_N describes how target pose change in camera coordinate with respect to the changing pose of the target itself in Σ_W .

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

$$\hat{\boldsymbol{\psi}}(t + \Delta t) = \hat{\boldsymbol{\psi}}(t) + \hat{\boldsymbol{\psi}}\Delta t = \hat{\boldsymbol{\psi}}(t) + \hat{\mathbf{J}}_A(\mathbf{q}, \hat{\boldsymbol{\psi}}) \begin{bmatrix} \dot{\mathbf{q}} & {}^W \dot{\boldsymbol{\phi}}_M \end{bmatrix}^T \Delta t. \quad (33)$$

(33) shows $\hat{\mathbf{J}}_A$ is a function of \mathbf{q} and $\hat{\boldsymbol{\psi}}$. Here we use $\hat{\boldsymbol{\psi}}$ since it is the result of recognition at time t by using model-based matching in which errors exist, derived from sensing dynamics, while \mathbf{q} and $\dot{\mathbf{q}}$ can be observable correctly from the robot manipulator. Then the errors are included in $\hat{\mathbf{J}}_A$ from $\hat{\boldsymbol{\psi}}(t)$ will lead to incorrect prediction and cause the tracking errors at the time $t + \Delta t$. It seems as a difficulty in 3-D pose prediction since the errors may increase drastically due to such a vicious circle as we can see in (33) that may amplify the errors in $\hat{\boldsymbol{\psi}}(t)$ to those of $\hat{\boldsymbol{\psi}}(t + \Delta t)$. However, the on-line optimization method, "1-Step GA" combined with the above prediction have an ability to prevent the pose tracking error from increasing by correcting the pose estimation through exploring nature of GA in heuristic searching behavior.

${}^W \dot{\boldsymbol{\phi}}_M$ expresses the velocity of the target object in the real world. This prediction part has been discussed widely by researches, for example,⁽⁸⁾ using kalman filter,⁽²²⁾ using neural network. Paper⁽²²⁾ presented our previous research that using a neural network system to predict the velocity of a swimming fish, which enabled a robot arm to catch fish without time delay.

In this paper, we do not deal with the two predictions shown (33) at one time. Since we have confirmed the second prediction: prediction based on the velocity of the target object in the real world in⁽²²⁾, this paper will mainly deal with the prediction of the target velocity in Σ_E based on the joint velocity $\dot{\mathbf{q}}$, so we rewrite (33) as

$$\hat{\boldsymbol{\psi}}(t + \Delta t) = \hat{\boldsymbol{\psi}}(t) + \hat{\mathbf{J}}_M(\mathbf{q}, \hat{\boldsymbol{\psi}})\dot{\mathbf{q}}\Delta t \quad (34)$$

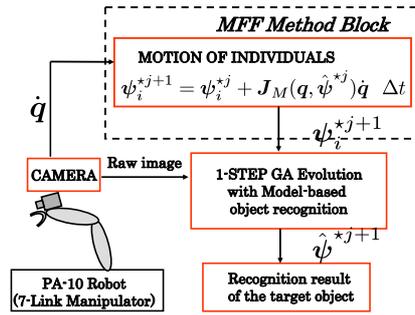


Fig. 6 Motion feed-forward tracking system

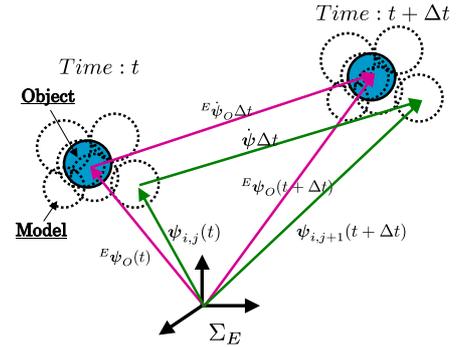


Fig. 7 MFF Compensation. Notice Σ_E and Σ_O are relative coordinates, here we suppose the end-effector is moving and the target is static.

3.2. “1-step GA + MFF” method

Here, we use p individuals for searching. The best one in p individuals in j -th generation at time $j\Delta t$ represented by $\psi_{max}^j(j\Delta t)$, is denoted as $\hat{\psi}^j(j\Delta t)$ from here, which represents the measured pose of the target object, $\hat{\psi}^j$ is described by

$$\hat{\psi}^j = \left\{ \psi_i^j(j\Delta t) \mid \max_{i=1,2,\dots,p} F(\psi_i^j(j\Delta t)) \right\}, \quad (35)$$

We define the individual of GA as ψ_i^{*j} in the case of using MFF method to predict to distinguish from ψ_i^j used in the case of not using MFF method. And the best gene in p individuals of ψ_i^{*j} is defined by $\hat{\psi}^{*j}$ to make a difference to $\hat{\psi}^j$. $\hat{\psi}^{*j}$ is described by

$$\hat{\psi}^{*j} = \left\{ \psi_i^{*j}(j\Delta t) \mid \max_{i=1,2,\dots,p} F(\psi_i^{*j}(j\Delta t)) \right\}. \quad (36)$$

Using the prediction of (33), the pose of the individuals ψ_i^{*j+1} in the next generation can be predicted based on the current pose $\hat{\psi}^{*j}$, presented by

$$\psi_i^{*j+1} = \psi_i^{*j} + \hat{J}_M(q, \hat{\psi}^{*j}) \dot{q} \Delta t, \quad (i = 1, 2, \dots, p). \quad (37)$$

The estimation system of the proposed method is shown in Fig. 6. The proposed MFF method can predict the motion of the target projected to the cameras based on the ego motion of the robot. So when the individuals of GA got converged, the whole group of genes ψ_i^{*j} , ($i = 1, 2, \dots, p$) will move together with the motion of the target in the image, never lose it even under a camera’s ego motion of robot manipulator. Thus, recognition by hand-eye cameras will be independent of the dynamical motion of the manipulator, then robust recognition can be expected as the same performance as using fixed cameras.

4. Controller

4.1. Desired-trajectory generation

The desired relative relationship of Σ_M and Σ_E is given by Homogeneous Transformation as ${}^{Ed}T_M(t)$, the difference of the desired camera pose Σ_{Ed} and the actual camera pose Σ_E is denoted as ${}^E T_{Ed}$. ${}^E T_{Ed}$ can be described by

$${}^E \hat{T}_{Ed}(t) = {}^E \hat{T}_M(t) {}^{Ed} T_M^{-1}(t), \quad (38)$$

Notice that (38) is a general deduction that satisfies arbitrary object motion ${}^W T_M(t)$ and arbitrary objective of visual servoing ${}^{Ed} T_M(t)$.

Differentiating (38) with respect to time yields

$${}^E \dot{T}_{Ed}(t) = {}^E \dot{T}_M(t) {}^{Ed} T_M^{-1}(t) + {}^E \hat{T}_M(t) {}^{Ed} \dot{T}_M^{-1}(t). \quad (39)$$

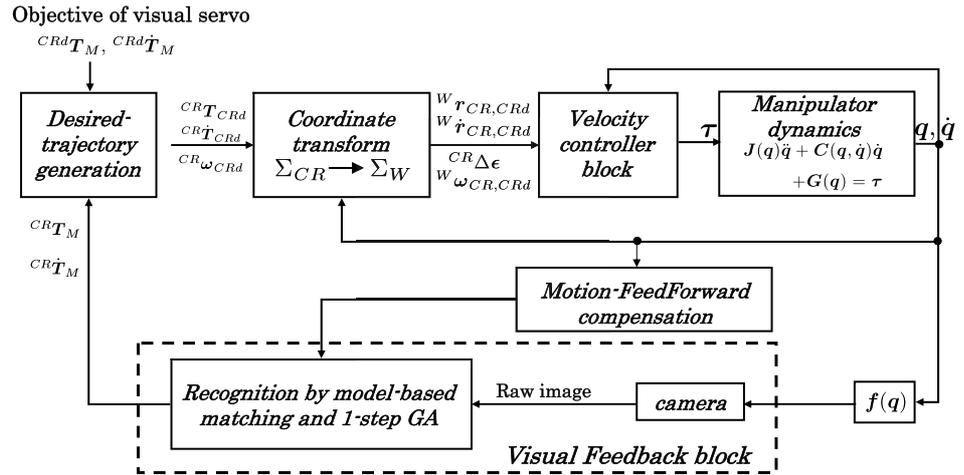


Fig. 8 Block diagram of the visual servoing system

Here, ${}^{Ed}T_M(t)$, ${}^{Ed}\dot{T}_M(t)$ are given as the desired visual servoing objective. ${}^E\hat{T}_M(t)$ is measured by cameras using the on-line recognition method proposed in Section 2. ${}^E\hat{T}_M(t)$ is then calculated by

$${}^E\dot{\hat{T}}_M(t) = ({}^E\hat{T}_M(t) - {}^E\hat{T}_M(t - \Delta t)) / \Delta t, \quad (40)$$

which is output periodically with a time of Δt regardless the object is moving or not.

4.2. 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. 8. Based on the above analysis of the desired-trajectory generation, the desired hand velocity ${}^W\dot{r}_d$ is calculated as,

$${}^W\dot{r}_d = K_{P_p} {}^W r_{E,Ed} + K_{V_p} {}^W \dot{r}_{E,Ed}, \quad (41)$$

where ${}^W r_{E,Ed}$, ${}^W \dot{r}_{E,Ed}$ are given by transforming ${}^E T_{Ed}$ and ${}^E \dot{T}_{Ed}$ from Σ_E to Σ_W . K_{P_p} and K_{V_p} are positive definite matrix to determine PD gain.

The desired hand angular velocity ${}^W\omega_d$ is calculated as,

$${}^W\omega_d = K_{P_o} {}^W R_E {}^E \Delta\epsilon + K_{V_o} {}^W \omega_{E,Ed}, \quad (42)$$

where ${}^E \Delta\epsilon$ is the quaternion error that from the recognition result directly, and ${}^W \omega_{E,Ed}$ can be calculated by transforming ${}^E T_{Ed}$ and ${}^E \dot{T}_{Ed}$ from Σ_E to Σ_W . Also, K_{P_o} and K_{V_o} are suitable feedback matrix gains.

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

$$\dot{q}_d = J^+(q) \begin{bmatrix} {}^W\dot{r}_d \\ {}^W\omega_d \end{bmatrix}. \quad (43)$$

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

$$\tau = K_{SP}(\dot{q}_d - \dot{q}) + K_{SI} \int_0^t (\dot{q}_d - \dot{q}) dt \quad (44)$$

where K_{SP} and K_{SI} are symmetric positive definite matrix to determine PI gain (Table 1).

5. 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].

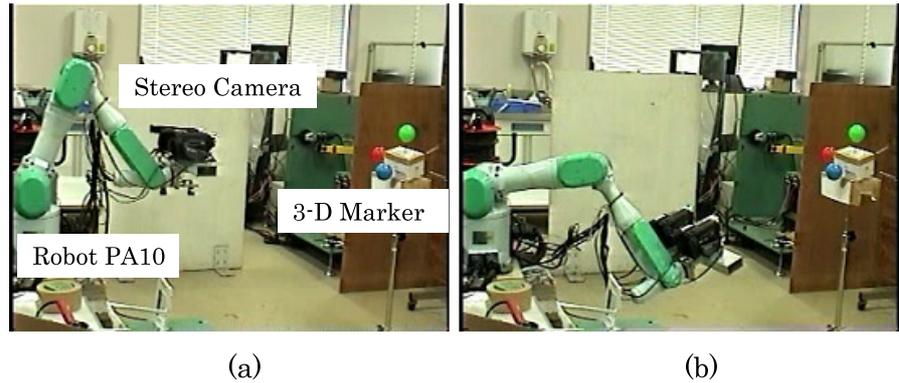


Fig. 9 Step response experiment. (a) Initial pose of Pa10. (b) Visual servoing to a static 3-D marker.

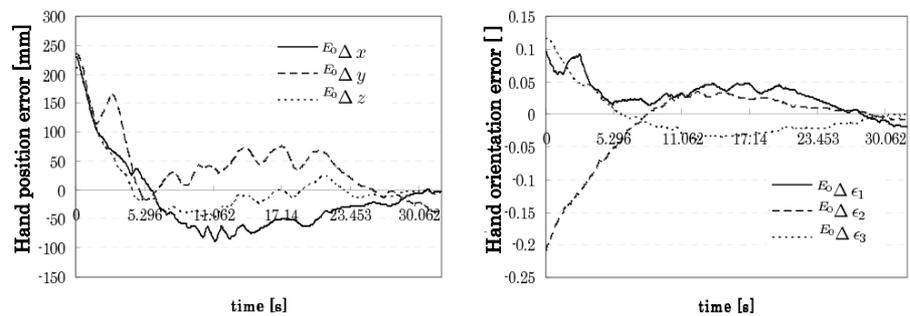


Fig. 10 Hand pose error of step response without using MFF

5.1. Experimental Condition

A photograph of our experimental system is shown in Fig. 15. 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 WORKSTATION PWS650 (CPU: Xeon, 2.00 GHz) host computer.

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

$${}^w T_{E_0} = \begin{bmatrix} 0 & 0 & 1 & -918 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 455 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (45)$$

position unit: [mm].

The parameters of GA is set as that shown in Table.2.

5.2. Experimental Results

5.2.1. Step Response Experiment Here, a static object is set as ${}^E \psi_M = [-70[mm], 70[mm], 1000[mm], 0.1, -0.2, 0.12]^T$, 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_E and Σ_M , as

$${}^E \psi_{Md} = [0[mm], 10[mm], 900[mm], 0, 0, 0]^T. \quad (46)$$

The initial pose of the robot manipulator is shown in Fig. 9(a), and the moved robot manipulator to satisfy ${}^E \psi_{Md}$ is shown in Fig. 9(b).

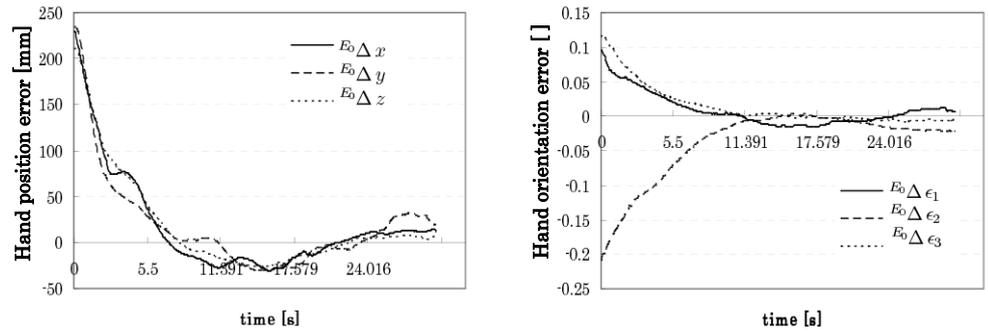


Fig. 11 Hand pose error of step response by using MFF

To show the effectiveness of the proposed MFF method, we perform the step response experiment with MFF method (that is “1-step GA + MFF” method) and without MFF method (that means using “1-step GA ” only) separately. Fig. 10 shows the difference of the desired hand pose and the actual hand pose in Σ_{E_0} without using MFF method. Fig. 11 shows the hand difference with using MFF method. Since we use quaternion to express the orientation of an object, so there is no orientation unit and we denote here as []. In Fig. 10, 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. 11, such vibrations existing in Fig. 10 had been suppressed. The end-effector position costs about 20[s] to converge to the desired position, and the orientation costs about 10[s] 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 3. By comparing the convergence speed with these researches, our system shows a good ability in visual servoing task.

5.2.2. 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.

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

$$\begin{cases} {}^{Ed}x_M(t) = 100\sin(\frac{2\pi}{T}t)[mm] \\ {}^{Ed}y_M(t) = 70[mm] \\ {}^{Ed}z_M(t) = 1300[mm] \\ {}^{Ed}\epsilon_{1M}(t) = 0 \\ {}^{Ed}\epsilon_{2M}(t) = 0 \\ {}^{Ed}\epsilon_{3M}(t) = 0 \end{cases} \quad (47)$$

The desired motion of the end-effector with respect to a static object is shown in Fig. 12.

First, we set the motion period of the manipulator T as 60[s]. Fig. 13 shows the errors of the desired hand pose and the actual hand pose in Σ_{E_0} without using MFF method. Fig. 14 shows the hand errors with using MFF method. The stereo cameras were shaking because

Table 1 Gain Parameters

Link Number	[L1 L2 L3 L4 L5 L6 L7]
K_{SP}	[3200 3200 1400 1400 1000 1000 1000]
K_{SI}	[1362 1362 596 596 596 426 426]

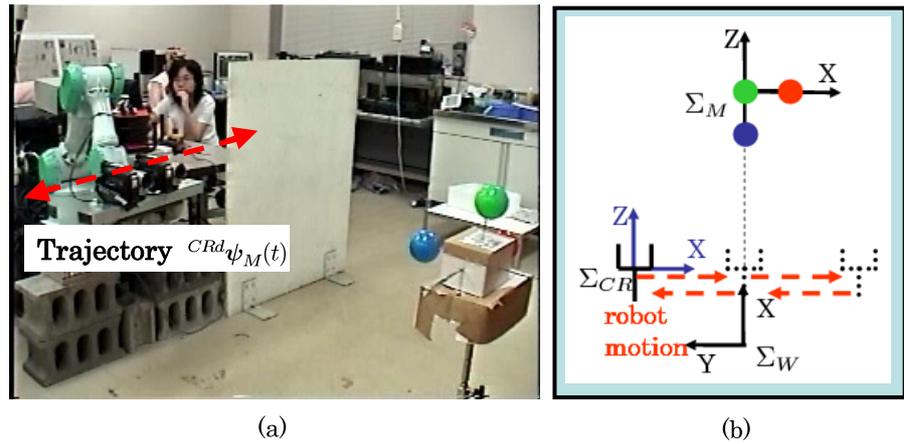


Fig. 12 (a)A photograph of time-varying visual servo. (b)Coordinate system of (a).

Table 2 GA Parameters

Population size	20 individuals
Selection rate	0.5
Crossover	Two-point
Mutation rate	0.10
Elitist model	yes

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. 13 indicated the system became unstable as time passing. Compared with Fig. 13, the errors in Fig. 14 were limited in a range, which means the system was under a stable control.

We compare the visual servoing with and without MFF 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 4 shows the mean value of the fitness function F defined as \bar{F} and the rms (root-mean-square) value of all components of ${}^{E_0}\tilde{\psi}$ defined as ${}^{E_0}\Delta\tilde{\psi} = [{}^{E_0}\Delta\tilde{x}, {}^{E_0}\Delta\tilde{y}, {}^{E_0}\Delta\tilde{z}, {}^{E_0}\Delta\tilde{\epsilon}_1, {}^{E_0}\Delta\tilde{\epsilon}_2, {}^{E_0}\Delta\tilde{\epsilon}_3]^T$ in each situation of $T = 60, 40, 20[s]$. We can see that without using MFF, \bar{F} gets lower, and ${}^{E_0}\Delta\tilde{\psi}$ gets bigger to about 96[mm] in $\Delta\tilde{x}_M$, and 0.045 in $\Delta\tilde{\epsilon}_{2M}$ (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 ${}^{E_0}\Delta\tilde{\psi}$ are not changed much, about 51[mm] in $\Delta\tilde{z}_M$ and 0.019 in $\Delta\tilde{\epsilon}_{2M}$ (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.

5.2.3. Visual Servoing To A Moving Object In this experiment, the target object is fixed on a mobile robot, as shown in Fig. 15. Fig. 16 shows the coordinate system corresponding

Table 3 Review of Literature

Reference	Convergence time of step response
(9)	about 9.9[s] when the desired position is parallel to the image plane, else, about 49.5[s].
(10)	in x,y, roll,pitch, yaw 30s, in z position about 70s
(6)	about 60s.
(7)	about 150s.
(23)	is about 200s.

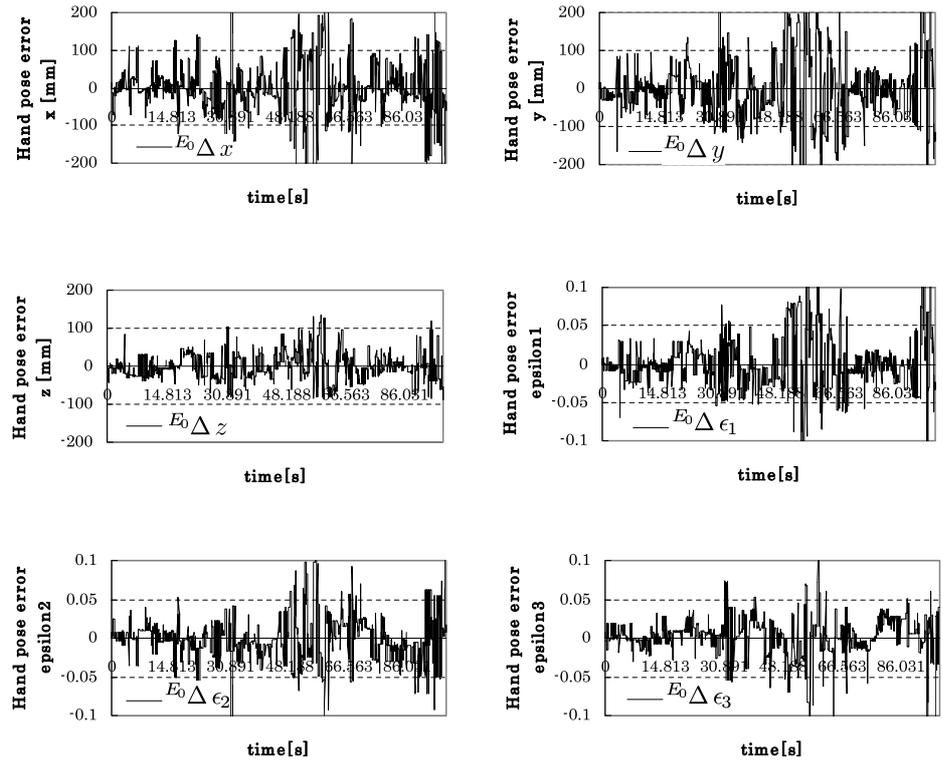


Fig. 13 Hand pose error of time-varying visual servoing without MFF method when $t = 60s$

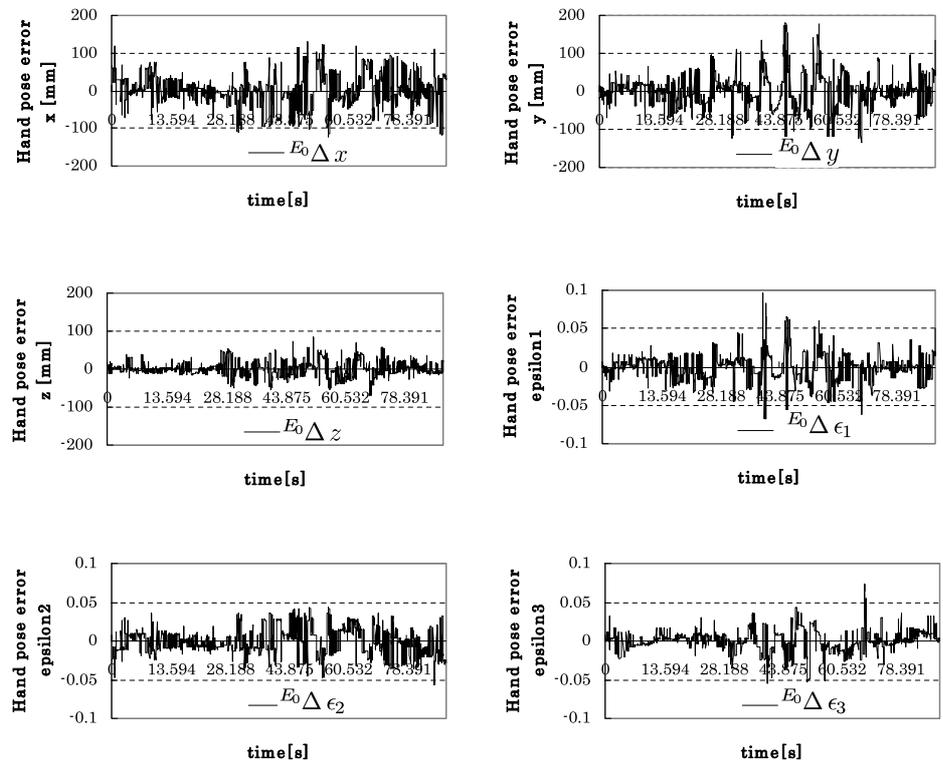


Fig. 14 Hand pose error of time-varying visual servoing with MFF method when $t = 60s$

to Fig. 15. 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

$$\theta_d[deg] = a \sin\left(\frac{2\pi}{T}\right)t, \quad (48)$$

Table 4 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

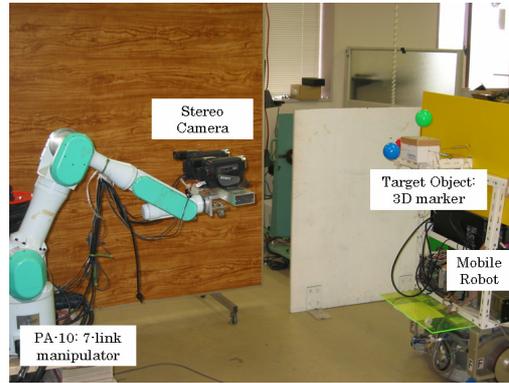


Fig. 15 A photograph of visual servo system

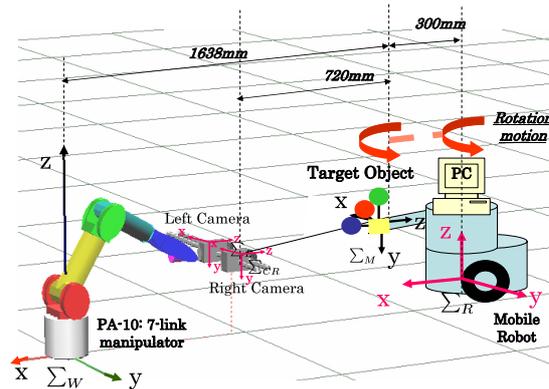


Fig. 16 Coordinate system of visual servoing

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

$$V_R = kp(\theta_d - \theta) + kv(\dot{\theta}_d - \dot{\theta}), \quad (49)$$

$$V_L = -V_R, \quad (50)$$

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_E , defined as ${}^{Ed} \psi_M = [0[mm], 10[mm], 700[mm], 0, 0, 0]^T$.

Figs. 17(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 ${}^{E0} \psi_E$, compared with the desired hand pose ${}^{E0} \psi_{Ed}$. Figs. 18(a) to (f) show the experimental results of ${}^{E0} \psi_E$ and ${}^{E0} \psi_{Ed}$ in the case of using MFF method. In the period of the trajectory of ${}^{E0} \psi_{Ed}$ 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. 17 and Fig. 18(a),(e) began to turn to curved line of sin/cos function, the mobile robot started to move. Comparing Figs. 17(a),(e) with Fig. 18(a),(e), the time-delay of hand motion in the case of using MFF method

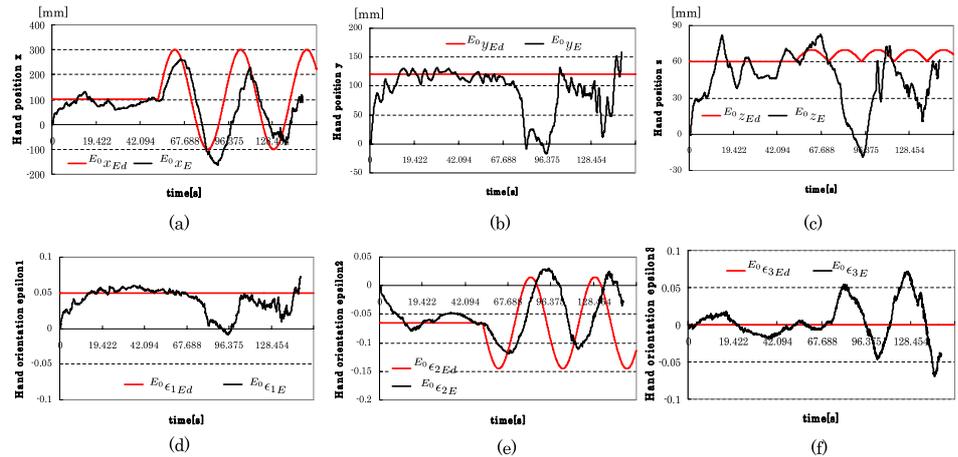


Fig. 17 Hand pose error of visual servoing without MFF method

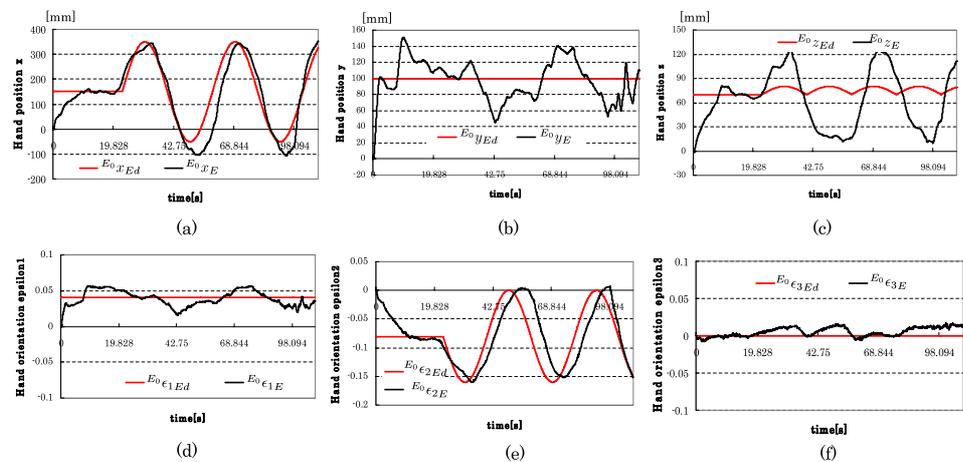


Fig. 18 Hand pose error of visual servoing with MFF method

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.

6. Conclusion

This paper deals with position-based 6-DoF visual servoing. We propose a MFF method to compensate the fictional 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. The on-line optimization method, “1-Step GA” is combined with MFF method to prevent the pose tracking error from increasing by correcting the pose estimation through exploring nature of GA in heuristic searching behavior. Visual recognition preciseness is improved, and the visual servoing become more stable by our proposal “1-step GA + MFF” method. The effectiveness of our proposed visual servoing system has been confirmed through experiments by a 7-link hand-eye manipulator.

References

- (1) A. A. Rizzi and D. E. Koditschek, *An Active Visual estimator for Dexterous Manipulation*, IEEE Trans. Robotics and Automation, Vol.12, No.5, pp.697-713, 1996.
- (2) J. A. Piepmeyer, G. V. McMurray, and H. Lipkin, *Uncalibrated Dynamic Visual Servoing*, IEEE Trans. Robotics and Automation, Vol.20, No.1, pp.143-147, 2004.
- (3) 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).
- (4) 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.

- (5) William J. Wilson, Carol C. Williams Hulls and Graham S. Bell, "Relative End-effector Control Using Cartesian Position Based Visual Servoing", IEEE Trans. on Robotics and Automation, vol. 12, No. 5, 1996, pp. 684-696.
- (6) 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.
- (7) 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.
- (8) 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.
- (9) 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.
- (10) 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.
- (11) R.Yang and Z.Zhang, "Model-based head pose tracking with stereovision", Proc.5th IEEE Int.Conf.on Automatic Face and Gesture Recognition, 2002.
- (12) S. Baker and I. Matthews, *Equivalence and efficiency of image alignment algorithms* in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, (Hawaii, USA), pp. 1090-1097, Dec 8-14, 2001.
- (13) H. Y. Shum and R. Szeliski, *Construction of panoramic image mosaics with global and local alignment*, International Journal of Computer Vision, vol. 16, no. 1, pp.63-84, 2000.
- (14) E. Mails, *Improving vision-based control using efficient second-order minimization techniques* in Proceedings of IEEE International Conference on Robotics and Automation, (New Orleans, USA), pp. 1843-1848, Apr. 6-May 1, 2004.
- (15) S. Benhimane and E. Mails, *Real-time image-based tracking of planes using efficient second-order minimization* in Proceedings of IEEE/RSJ Conference on Intelligent Robot and Systems, (Sendai, Japan), pp. 1090-1097, Sep 28-Oct 2, 2004.
- (16) Y. Keller and A. Averbuch, *Fast motion estimation using bidirectional gradient methods*, IEEE Transactions on Image Processing, vol. 13, no. 8, pp. 1042-1054, 2004.
- (17) 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.
- (18) 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.
- (19) 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).
- (20) B.Siciliano and L.Villani: *Robot Force Control*, ISBN 0-7923-7733-8.
- (21) M. MINAMI, J. ZHU and M. MIURA, *Real-time Evolutionary Recognition of Human with Adaptation to Environmental Condition*, The Second Int. Conf. on Computational Intelligence, Robotics and Autonomous Systems (CIRAS 2003).
- (22) M. Minami and T. Yoshida "Prediction Servoing to Catch Escaping Fish Using Neural Network", Proceedings of the 2008 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics July 2 - 5, 2008, Xi'an, China, pp1225-1231
- (23) 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.