Experimental Evaluations of Prediction Servoing to Moving Object by Hand-eye Robotic System

Yusuke Sunami[1], Mamoru Minami[1], Akira Yanou[1]

[1]Graduate School of Natural Science and Technology, Okayama University, Okayama, Japan
(Tel: +81-86-251-8924; E-mail: (y-sunami,minami,yanou)@suri.sys.okayama-u.ac.jp)

Abstract: In the field of robot vision, a control method called visual servoing attracts attention. The visual servoing is a method to control robots by visual information in a feedback loop, which is obtained by video cameras. So, this method is expected to be able to make robots adapt to tasks in changing or unknown environment. However, when the target object moves quickly, it happens to be unable for the robots to track it due to dynamical effect, i.e., motion delay. To decrease the delay, we have proposed prediction servoing control method, which utilizes prediction of the target position based on the past observed position data of the object and learning by neural networks, and utilize predicted position as a desired position for the visual servoing. In this research, we have confirmed how the learnig function in neural networks work for precise prediction of target's future position through visual servoing experiments.

Keywords: Visual Feedback Control, GA, Prediction Servoing.

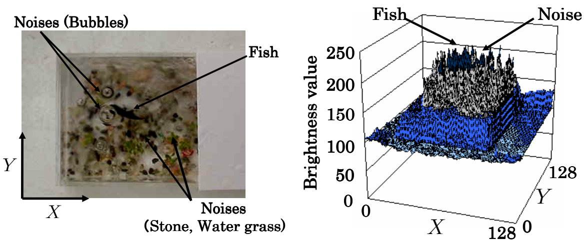
1. INTRODUCTION

Nowadays, the robot has been used in various fields. In recent years, robots' behavior is required some ability under the unknown environment and autonomous operation. However, robots' behaviors are generally required to repeat a sequence of motions and need the sequence of motion to be taught by operators through commands written by robotic language. However, robots, working in a disaster area, space, hospital, and home, are required to automatically work in unknown environment. Therefore, we need a visual servoing system to recognize surrounding unknown environment. Visual servoing is thought to be useful for making the robot active in unknown environment and in constantly changing environment, because it can control the robot using feedback loop built visual information obtained from visual sensors.

However, when the target object moves quickly, it happens to be unable for the robot to track it, because of its motion delay derived from dynamics of robots. To decrease the delay time, we have proposed a prediction servoing control method, which is the method of predicting by neural networks(N.N.) the position of the target object based on the past position data of the object and learning ability of the N.N. to decrease predicted errors, and utilize it as a desired position for the visual servoing[1]-[5].

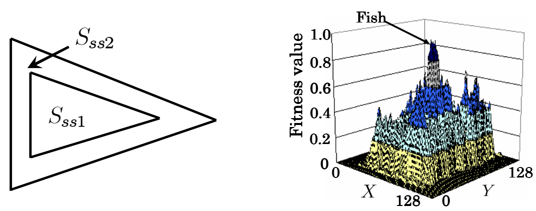
The method that predict future target position uses circular approximation—calculate a future position on a presumption of circular orbit based on the past target position information. However, the target object does not necessarily move along circular orbit. Error sometime unavoidably increase when the circular approximation method is only used. So that, we use neural networks(N.N.) to reduce the prediction error to be zero.

In this research, the experiments were done to contrast visual servoing with prediction servoing. The superiority of prediction servoing was analyzed, and the effectiveness of it was confirmed.



(a)2-D raw-image (b)3-D raw-image

Fig. 1 Raw-image of swimming fish



(a)Surface-strips model (b)$F(\boldsymbol{\phi})$

Fig. 2 Surface-strips Model to search a fish

2. TRACKING METHOD

The tracking method of a target object uses Model-based Matching method, 1-step GA and Gazing GA.

The model-based Matching method is a evaluation method by fitness function value $F(\boldsymbol{\phi})$. $F(\boldsymbol{\phi})$ is determined by the degree of overlapping of the model and the target object —the model has information of shape and brightness value of the target object. And to search for such a target fish in the raw-image, a geometrical triangular shape of the surface-strips model as shown in Fig.2(a) is used. Let us denote a set of the coordinates inside the surface of the model as S_{ss1} and the contour-strips S_{ss2}, and the combination as S. Here, the correlation function $F(\boldsymbol{\phi})$ used as fitness function of the surface-strips model with the image is given as Eq.(1).

$$F(\boldsymbol{\phi}(t)) = \sum_{\boldsymbol{r}(t)\in S_{ss1}(\boldsymbol{\phi}(t))} p(\boldsymbol{r}(t)) - \sum_{\boldsymbol{r}(t)\in S_{ss2}(\boldsymbol{\phi}(t))} p(\boldsymbol{r}(t)) \quad (1)$$

It gives low fitness function value when the target object's shape is different from the model, giving high function value only when shape and brightness value of the target object and the model match. We get the correct information of the target object's pose.

1-step GA scatters the individual data with the pose of the target object in the image, and finds target object's pose using selection, crossover and mutation. And evolutionary process for the GA processing in real time is repeated within a time interval of video rate(33fps) in the Fig. 3.

Gazing GA adjusts the range of mutation by the level of the fitness function value. Gazing GA searches globally when fitness function is low value and do locally when fitness function is high value in the Fig. 4,5.

Detail of these method is written in references[5]-[9].

Table 1 Parameter of GA

$Populationsize$	$40individuals$
$Selectionrate$	0.4
$Mutationrate$	0.1
$Lengthperindividual$	$12bits$
$Elitistmodel$	yes

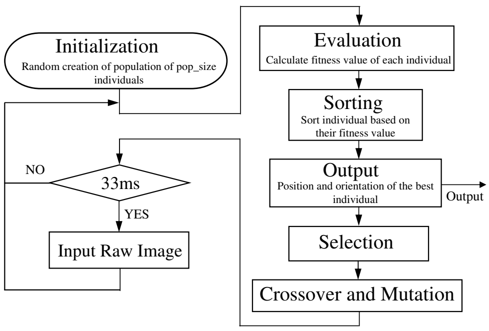


Fig. 3 1-step GA

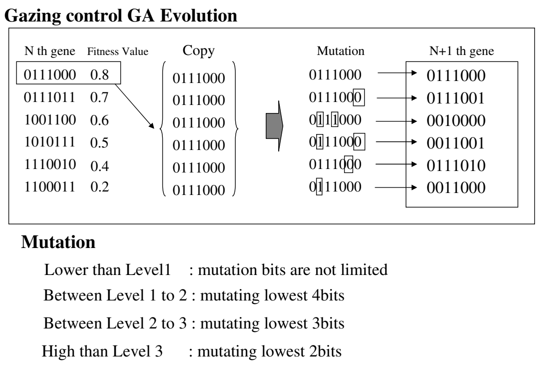


Fig. 4 Gazing GA

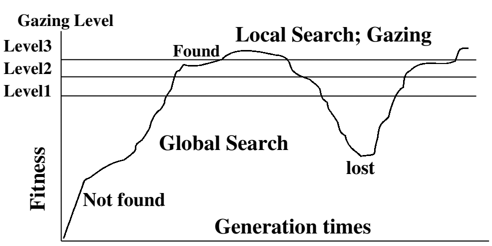


Fig. 5 Gazing switching

3. CIRCULAR APPROXIMATION

As first step for deriving a predicted future position of the target object, the target object's movement is assumed to be circular motion, calculating the center point, radius, and angular velocity of the circle from the position data of the past successive three points.

Figure6 shows the method to calculate the center point of the circle from the position data of the three points. Here, $\boldsymbol{r}_{n-2} = (x_{n-2}, y_{n-2})$ and $\boldsymbol{r}_{n-1} = (x_{n-1}, y_{n-1})$ denote past target objects' position coordinates, and $\boldsymbol{r}_n = (x_n, y_n)$ does a current target object's position coordinate, and $\hat{\boldsymbol{r}}_{n+k} = (\hat{x}_{n+k}, \hat{y}_{n+k})$ does a future target object's position coordinate at future time of $\Delta t \cdot (n+k)$[s]. The center point of the circle denoted by $\boldsymbol{p}_n = (p_n, q_n)$ at time t, where n denotes current serial number and $\Delta t \cdot n$ means current time t. Therefore, radius l_n can be calculated by

$$l_n = |\boldsymbol{r}_n - \boldsymbol{p}_n|. \tag{2}$$

Assuming that the fish moves on a circle in successive three calculation time, we have

$$|\boldsymbol{r}_n - \boldsymbol{p}_n| = |\boldsymbol{r}_{n-1} - \boldsymbol{p}_n| = |\boldsymbol{r}_{n-2} - \boldsymbol{p}_n|. \tag{3}$$

The center position coordinates $\boldsymbol{p}_n = (p_n, q_n)$ is calculated by the Eq.(3), and the solutions are below,

$$\begin{aligned} p_n = \frac{Y_{10}}{2(X_{10})(Y_{21}) - 2(X_{21})(Y_{10})} \Bigg\{ \left(\frac{Y_{21}}{Y_{10}}\right) \\ (x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2) \\ -(x_{n-2}^2 - x_{n-1}^2 + y_{n-2}^2 - y_{n-1}^2) \Bigg\} \end{aligned} \tag{4}$$

$$\begin{aligned} q_n = \frac{1}{2(Y_{21})}(x_{n-2}^2 - x_{n-1}^2 \\ +y_{n-2}^2 - y_{n-1}^2 - 2X_{21}p_n). \end{aligned} \tag{5}$$

Where,

$$X_{21} = x_{n-2} - x_{n-1}, X_{10} = x_{n-1} - x_n$$
$$Y_{21} = y_{n-2} - y_{n-1}, Y_{10} = y_{n-1} - y_n.$$

Also radius l_n can be calculated from the Eq.(2). Here, we denote $\boldsymbol{\rho}_n = \boldsymbol{r}_n - \boldsymbol{p}_n$ and $\Delta\boldsymbol{\rho}_n = \boldsymbol{\rho}_n - \boldsymbol{\rho}_{n-1}$, and consider the equation shown below,

$$\boldsymbol{t} = \boldsymbol{\rho}_n \times \Delta\boldsymbol{\rho}_n. \tag{6}$$

Using z-component of the vector $\boldsymbol{t} = [t_x, t_y, t_z]^T$, the value of angular velocity can be approximated by the covering distance by adopting radius and point $n-1$ to point n as

$$\omega_n \approx sign(t_z) \frac{|\boldsymbol{\rho}_n - \boldsymbol{\rho}_{n-1}|}{\Delta t \cdot l_n}. \tag{7}$$

In order to calculate a future position of the target object from the angular velocity and radius, target objects' position coordinates must be represented by a trigonometric. A target object's position $\boldsymbol{r}_n = (x_n, y_n)$ at current time n is represented by Eq.(9), (10), $\boldsymbol{p}_n = (p_n, q_n)$ is calculated by target position coordinates of three points and α expresses the angle of the current object's position based on the horizontal line shown in the Fig.6,7.

$$\alpha = atan2(y_n - q_n, x_n - p_n) \quad (8)$$
$$x_n = p_n + l_n \cos\alpha \quad (9)$$
$$y_n = q_n + l_n \sin\alpha \quad (10)$$

Then a predicted future position $\hat{\boldsymbol{r}}^C_{j+k} = (\hat{x}^C_{j+k}, \hat{y}^C_{j+k})$ through circular approximation after $\Delta t \cdot k$[s] from current time t is calculated by

$$\hat{x}^C_{n+k} = p_n + l_n \cos(\alpha + k\omega_n \Delta t) \quad (11)$$
$$\hat{y}^C_{n+k} = q_n + l_n \sin(\alpha + k\omega_n \Delta t). \quad (12)$$

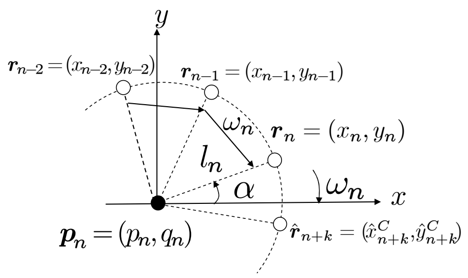


Fig. 6 Circular Approximation by using vector

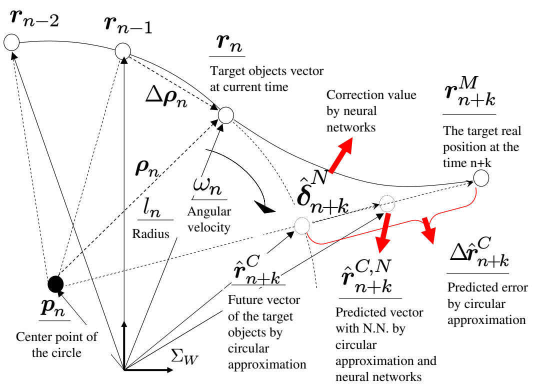


Fig. 7 Circular approximation by using N.N.

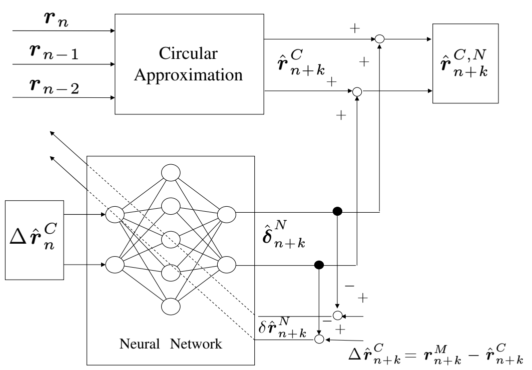


Fig. 8 Block diagram of prediction method

4. PREDICTION LEARNING

In this research, the neural network learning is given by the error Back Propagation(B.P.) [1][2]. The error Back Propagation is one of the neural network learning method—it has been highly regarded as it has a non-linear function approximation ability through adjusting connection weights minutely. The neural network changes the input-output relation by adjusting the values of connection weights w_{ij} and thresholds θ_i of the neurons that make up the network.

4.1 Neural network to correct the Circular Approximation Error

When a predicted future target object position $\hat{\boldsymbol{r}}^C_{n+k}$ —based on circular approximation— and an actual target position $\boldsymbol{r}^M_{n+k}$ at the time $\Delta t \cdot (n + k)$ are exactly the same, the circular approximation error is 0. And it means that the prediction is correctly working. However, in fact, it may happen predicted error $\Delta\hat{\boldsymbol{r}}^C_{n+k}$ at the time $\Delta t \cdot (n + k)$ because the target object behavior is non-linear and emotional. The error is,

$$\Delta\hat{\boldsymbol{r}}^C_{n+k} = \boldsymbol{r}^M_{n+k} - \hat{\boldsymbol{r}}^C_{n+k}. \quad (13)$$

Here, the predicted error after the time $\Delta t \cdot k$ has passed from current time $\Delta t \cdot n$ is defined as E_{n+k}.

$$E_{n+k} = \|\Delta\hat{\boldsymbol{r}}^C_{n+k}\| \quad (14)$$

The block diagram of the future target object's position prediction using in N.N. being learned by Back Propagation(B.P.) is shown in Fig.8. First, $\hat{\boldsymbol{r}}^C_{n+k}$ is calculated by circular approximation by using Eq.(11),(12). The input of the N.N. at the time $\Delta t \cdot n$ is the predicted error vector $\Delta\hat{\boldsymbol{r}}^C_n$ at the time $\Delta t \cdot (n - k)$, that is

$$\Delta\hat{\boldsymbol{r}}^C_n = \boldsymbol{r}^M_n - \hat{\boldsymbol{r}}^C_n. \quad (15)$$

As shown in the Fig.8, the output of N.N. is $\hat{\boldsymbol{\delta}}^N_{n+k}$. And a prediction position with correction by N.N. $\hat{\boldsymbol{r}}^{C,N}_{n+k} = (\hat{x}^{C,N}_{n+k}, \hat{y}^{C,N}_{n+k})$ is denoted by Eq.(16).

$$\hat{\boldsymbol{r}}^{C,N}_{n+k} = \hat{\boldsymbol{r}}^C_{n+k} + \hat{\boldsymbol{\delta}}^N_{n+k} \quad (16)$$

The block diagram in Fig.8 expresses that we can calculate prediction position of the target object $\hat{\boldsymbol{r}}^{C,N}_{n+k}$ by using circular approximation $\hat{\boldsymbol{r}}^C_{n+k}$ and correction with N.N. by $\hat{\boldsymbol{\delta}}^N_{n+k}$. The teaching signal of N.N. $\delta\boldsymbol{r}^N_{n+k}$ is defined as

$$\delta\hat{\boldsymbol{r}}^N_{n+k} = \Delta\hat{\boldsymbol{r}}^C_{n+k} - \hat{\boldsymbol{\delta}}^N_{n+k}. \quad (17)$$

N.N. learns to diminish $\delta\boldsymbol{r}^N_{n+k}$ to $\mathbf{0}$ by Error Back Propagation method. Then given the situation that we can assume $\delta\hat{\boldsymbol{r}}^N_{n+k} \to \mathbf{0}$ when N.N. learning has been completed successfully. Then, $\Delta\hat{\boldsymbol{r}}^C_{n+k} = \hat{\boldsymbol{\delta}}^N_{n+k}$ is derived from Eq.(17). Then Eq.(13) is rewritten as

$$\hat{\boldsymbol{\delta}}^N_{n+k} = \boldsymbol{r}^M_{n+k} - \hat{\boldsymbol{r}}^C_{n+k}. \quad (18)$$

By substituting Eq.(18) to Eq.(16), we get

$$\hat{\boldsymbol{r}}^{C,N}_{n+k} = \boldsymbol{r}^M_{n+k}. \quad (19)$$

This means that predicted position using circular approximation with correction by N.N. converges to the position of the object at the time $\Delta t \cdot (n + k)$ if N.N. should decrease $\delta\hat{\boldsymbol{r}}^N_{n+k}$ to zero by Back Propagation Learning.

$\Delta\hat{\boldsymbol{r}}^C_n$ is difference between position of the target object $\boldsymbol{r}^M_n$ and predicted future position by the circular approximation $\hat{\boldsymbol{r}}^C_n$ in the previous-time at $\Delta t \cdot (n - k)$[s] .

And $\hat{\boldsymbol{\delta}}^N_{n+k}$ is an estimated error that is difference between future position of the target object $\boldsymbol{r}^M_{n+k}$ and $\hat{\boldsymbol{r}}^C_{n+k}$—is a prediction future position by the circular approximation in the future time $\Delta t \cdot (n+k)$[s]. This means that this method shows that we get correct position of the target object when using both circular approximation and neural networks.

This method means if $|\delta\hat{\boldsymbol{r}}^N_{n+k}| < \epsilon$ then guarantees that $|\hat{\boldsymbol{r}}^{C,N}_{n+k} - \boldsymbol{r}^M_{n+k}| < \Delta(\epsilon)$. That means if the input for N.N. $\delta\hat{\boldsymbol{r}}^N_{n+k}$ be kept to be less than ϵ by N.N.'s learning ability, then the prediction error $|\hat{\boldsymbol{r}}^{C,N}_{n+k} - \hat{\boldsymbol{r}}^M_{n+k}|$ will be kept within $\Delta(\epsilon)$ that dependently be determined by ϵ .

5. EXPERIMENTS

5.1 Experiment environment

Summary of the experimental machine is shown in Fig.9. Arm robot PA-10's hand—made by Mitsubishi Heavy Industries— attaches video camera —video rate 33[fps].

And we use DELL Optiplex(CPU:Pentium42GHz) to control the robot arm and predict behavioral and image recognition, image input boards are used the cybertech CT3001Rev2. It search in the GA for input image data from the video camera. Then, obtained the recognition position is saved and do as trajectory data. The target object moves at an average of 10 seconds of 920mm length course that can be linear motion and circular motion, it shows in the Fig.10. And shape of the object is a black triangle object height 35mm, length 20mm.

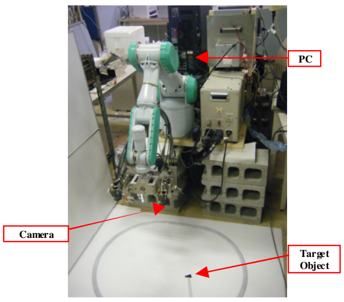


Fig. 9 Photograph of experiment system

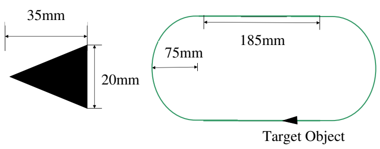


Fig. 10 Target object and its moving course

5.2 Visual Servoing

The first, visual servo tracking control with queuing is explained as follows. The relationship between the robot and camera coordinates in the visual servo system incorporating a video recognition Gazing GA method shows in Fig.11. The range area of the camera view is 150 and 120[mm] in x and y directions. The video recognition Gazing GA system consists of 1-step GA[10]. The best recognition results at each time $\underline{\boldsymbol{\phi}}_n = [x^{GA}_n, y^{GA}_n]$ represents the model's position—based on camera coordinate Σ_C— at the time $\Delta t \cdot n$ in pixels. For this reason, the real target object's position $\boldsymbol{r}^M_n$ expresses ${}^C\boldsymbol{r}^M_n$. Then position error between the target object and the center position of camera image can be described as follow,

$$\Delta^C\boldsymbol{r}_n = {}^C\boldsymbol{r}^M_n - {}^C\boldsymbol{r}^H_n \tag{20}$$

$${}^C\boldsymbol{r}^M_n = \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \underline{\boldsymbol{\phi}}_n = \begin{bmatrix} k_x x^{GA}_n \\ k_y y^{GA}_n \end{bmatrix}. \tag{21}$$

The origin of the camera and recognition position are to match so that it is ${}^C\boldsymbol{r}^H_n = \boldsymbol{0}$. k_x, k_y are coefficients to convert to unit[mm] to unit[pixel] in the image. k_x, k_y value are determined by the zoom factor of the input image and the mounting position of the camera of the experimental system. To move the manipulator in the direction of a object, position error $\Delta^C\boldsymbol{r}_n = [\Delta^C x_n, \Delta^C x_n]$ that represented camera coordinate in unit[pixel] need to change to work coordinate $\Delta^W\boldsymbol{r}_n$ in unit[mm] in coordinate Σ_W. Here, its conversion equation denotes Eq.(22).

$$\Delta^W\boldsymbol{r}_n = {}^W\boldsymbol{R}_C \Delta^C\boldsymbol{r}_n \tag{22}$$

Here, ${}^W\boldsymbol{R}_C$ is the pose transformation matrix with Σ_W and Σ_C. Robot hands' speed reading is described as follow,

$$\dot{\boldsymbol{r}}^d_n = \boldsymbol{K}_P \Delta^W\boldsymbol{r}_n + \boldsymbol{K}_V(\Delta^W\boldsymbol{r}_n - \Delta^W\boldsymbol{r}_{n-1}). \tag{23}$$

The desired joint variable $\dot{\boldsymbol{q}}^d_n$ is determined by inverse kinematics from $\dot{\boldsymbol{r}}^d_n$ by using the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q})$, and it is expressed by

$$\dot{\boldsymbol{q}}^d_n = \boldsymbol{J}^+(\boldsymbol{q})\dot{\boldsymbol{r}}^d_n \,, \tag{24}$$

where $\boldsymbol{J}^+(\boldsymbol{q})$ is the pseudo-inverse matrix of $\boldsymbol{J}(\boldsymbol{q})$. The robot used in this experimental system is a 7-Link manipulator, Mitsubishi Heavy Industries PA-10 robot. The control system, based on a PI control of PA-10 is expressed as

$$\boldsymbol{\tau} = \boldsymbol{K}_{SP}(\dot{\boldsymbol{q}}^d_n - \dot{\boldsymbol{q}}_n) + \boldsymbol{K}_{SI}(\boldsymbol{q}^d_n - \boldsymbol{q}_n). \tag{25}$$

Here, $\boldsymbol{q}^d_n$ is expressed as follow,

$$\boldsymbol{q}^d_n = \boldsymbol{f}^{-1}(\boldsymbol{r}^d_n). \tag{26}$$

$\boldsymbol{f}^{-1}()$ means the equation of motion of the inverse kinematics.

Here, $\boldsymbol{K}_{SP}$ and $\boldsymbol{K}_{SI}$ are positive definite diagonal matrix. If target angular velocities $\boldsymbol{q}^d_n$ converge to a constant value by GA search, this mean, when target object stop and GA converge to the pose of the target object, goal position $\boldsymbol{r}^d_n$ correspond to target objects position, and the robot hand is controlled directly above the object. Servoing system that incorporate the controller of Eq.(25) shows in Fig.12. Here, gain $\boldsymbol{K}_P, \boldsymbol{K}_V, \boldsymbol{K}_{SP}, \boldsymbol{K}_{SI}$ of Eq.(23),(25) denote Table 2.

5.3 Prediction Servoing

We explain prediction servoing that predict the future position of the target object and control the minions of the robot hand to the predicted position. In visual servoing, servoing of the target object was done using $\Delta^W\boldsymbol{r}_n$

Table 2 Gain parameters

$\boldsymbol{K}_P$	[0.75 0.75]
$\boldsymbol{K}_V$	[0.35 0.35]

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

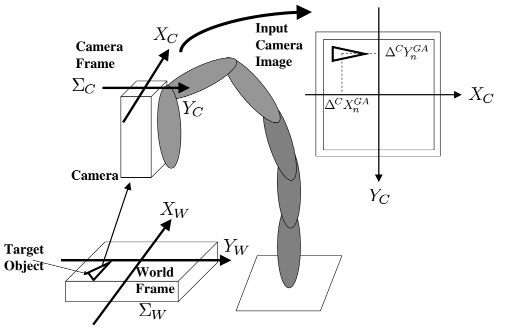


Fig. 11 Coordinates conversion

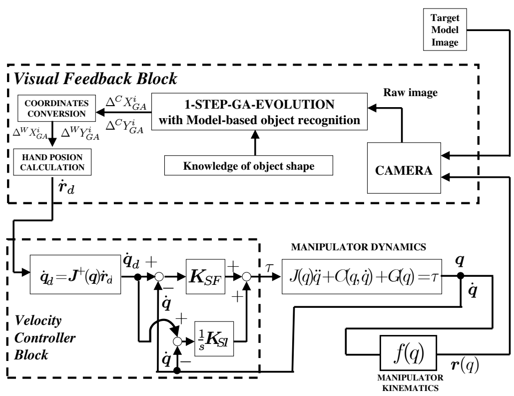


Fig. 12 Block diagram of the controller

—position error between the object and the hand. However, when the target object moves quickly, it happens to be unable for the robot hand to track it due to robot's motion delay.

Here, we considered control that can track an object can be performed while anticipating when we can build a system to track the prediction position using predicted future position $\hat{\boldsymbol{r}}^{C,N}_{n+k}$ of the target object in circular approximation and neural networks $\Delta^W \boldsymbol{r}_n$ is position error of the target object in Eq.(22). Therefore, Eq.(22) will be speed reading about predicted position when predicted position of the target object. And it can change goal velocity $\dot{\boldsymbol{r}}^d_n$ in visual servoing to prediction servoing. It will be noticed that $\Delta^W \boldsymbol{r}_n$ is used in Eq.(27) for velocity feedback. Therefore, Eq.(23) denotes Eq.(27) in prediction servoing change $\Delta^W \boldsymbol{r}_n$ to predicted error $\Delta^W \hat{\boldsymbol{r}}^{C,N}_{n+k}$.

$$\dot{\boldsymbol{r}}^d_n = \boldsymbol{K}_P \Delta^W \hat{\boldsymbol{r}}^{C,N}_{n+k} + \boldsymbol{K}_V (\Delta^W \boldsymbol{r}_n - \Delta^W \boldsymbol{r}_{n-1}) \quad (27)$$

Here,

$$\Delta^W \hat{\boldsymbol{r}}^{C,N}_{n+k} = {}^W\boldsymbol{R}_C \Delta^C \hat{\boldsymbol{r}}^{C,N}_{n+k} \quad (28)$$

$$\Delta^C \hat{\boldsymbol{r}}^{C,N}_{n+k} = {}^C\hat{\boldsymbol{r}}^{C,N}_{n+k} - {}^C\boldsymbol{r}^H_n. \quad (29)$$

Since this term is aimed to work for stabilizing the robot hand and reducing the hand oscillation derived term, $\Delta^W \boldsymbol{r}_n$ is used instead of $\Delta^W \hat{\boldsymbol{r}}^{C,N}_{n+k}$. $\boldsymbol{K}_P, \boldsymbol{K}_V, \boldsymbol{K}_{SP}, \boldsymbol{K}_{SI}$ are set at same variables in Table 2.

5.4 Comparison of prediction servoing and visual servoing in experiments

In order to verify whether it is valid prediction servoing, we compared it with the visual servoing. In this experiment, we use the target object that is moving at an average of 10 seconds of 920mm length course that can be linear motion and circular motion, and the shape is a black triangle object height 35mm, length 20mm, it shows in the Fig.10. Servoing experiment is carried out about 160[s] both visual servoing, prediction servoing.

The data that is got at the experiment by each method denotes Fig.13-16. Figure 13,14 are the image of every seconds from camera image each method. From this result, we confirmed that the target object is in the center position of the camera by prediction servoing than visual servoing. Figure 15,16 show the norm the target object's position how far from the center position of the camera image, and this error denotes $|\Delta^C \boldsymbol{r}_n|$. Average error $|\Delta^C \boldsymbol{r}_n|$ in visual servoing is 37.9[mm], and it in prediction servoing is 19.3[mm]. From this result, we confirmed that prediction servoing is more sensitive than visual servoing.

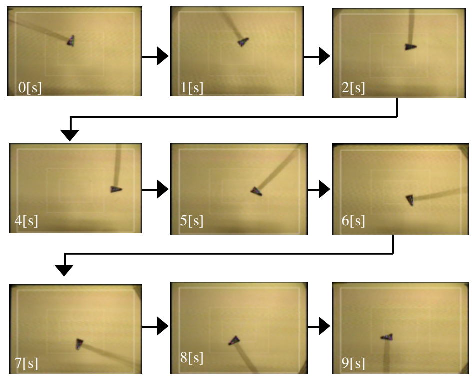


Fig. 13 Target object and hand motion image in visual servoing

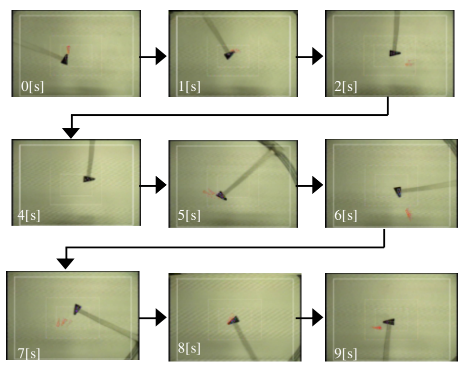


Fig. 14 Target object and hand motion image in prediction servoing

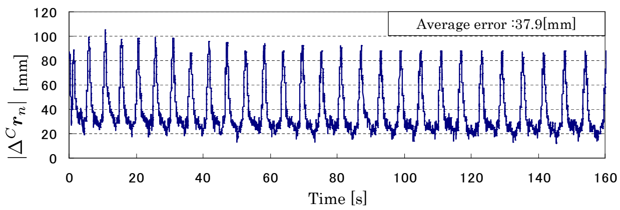


Fig. 15 Target object and hand error in visual servoing

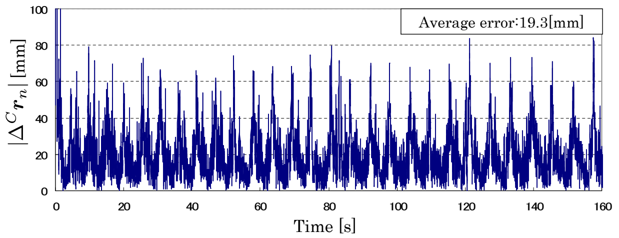


Fig. 16 Target object and hand error in prediction servoing

5.5 Change of the prediction error caused by changes in the neural network

We looked in any changes in the neural network what is happening with the prediction servoing. We confirmed the total sum of the coefficients in the neural network has decreased from Fig.17. The vertical axis in the Fig.17 denotes input-output ratio $|\hat{\boldsymbol{\delta}}_n^N|/|\Delta\hat{\boldsymbol{r}}_n^C|$ in the neural network. $|\Delta\hat{\boldsymbol{r}}_n^C|$ is a norm of the error the input of in the neural network—refer Eq.(15), and $|\hat{\boldsymbol{\delta}}_n^N|$ is a norm of correction value by the neural network. We confirmed the error $|\Delta^C\hat{\boldsymbol{r}}_n^{C,N}|$ —predicted future position $\hat{\boldsymbol{r}}_n^{C,N}$ using neural networks and circular approximation and real recognize position ${}^C\boldsymbol{r}_n^M$— decreases every time elapsed, as shown in Fig.18. Table.3 shows average error of the first half and the second half, it explain that we got better servoing result at second half than first. In this result, we confirmed the neural network learning is that it allows the goal to reduce the steady state error.

In this experiment, a target object moves a fixed course. We did other experiment when a target object is a fish—it moves random. As the result, it is better result by prediction servoing method than visual servoing method. Detail is written in references[5][11].

6. CONCLUSION

In this research, we confirmed that we got more sensitive results by prediction servoing using circular approximation and neural network than visual servoing, when robot hand servo the target object that repeat the motion of certain. Error Back Propagation that is learning method of neural network is learning to reduce the stationary error every time passes, We confirmed the effectiveness of the neural network. As future issues and goal, now we used only one camera to recognition, so we improve the performance of the object recognition using two cameras. Also we make experiment systems to recognize three-dimensional control because it is not only two-dimensional recognition.

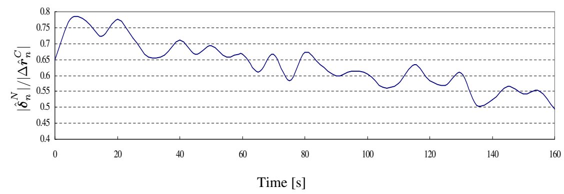


Fig. 17 Output/Input in neural network

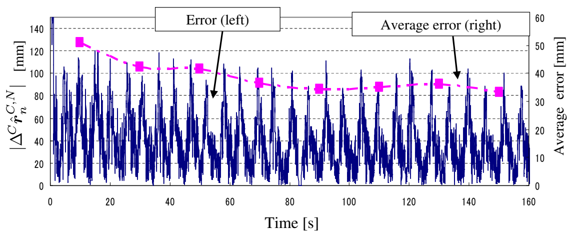


Fig. 18 Error that compared real position with prediction position in prediction servoing

Table 3 Average error that compared real position with prediction position in prediction servoing

	0-80[s]	80-160[s]
$\lvert\Delta^C\hat{x}^{C,N}\rvert$[mm]	31.4	24.9
$\lvert\Delta^C\hat{y}^{C,N}\rvert$ [mm]	24.0	19.3

REFERENCES

[1] On-line Prediction of Escaping Fish from Catching Net by Neural Network and Circular Approximation Toshiaki Yoshida, Mamoru Minami, Yasushi Mae, Hidekazu Suzuki Proceedings of SICE-ICCAS International Joint Conference 2006, SP04-1, A1528

[2] Fish Catching by Visual Servoing using Neural Network Prediction Toshiaki Yoshida, Mamoru Minami and Yasushi Mae SICE Annual Conference 2007, pp. 2372-2378

[3] Learning of Fish Movement Pattern by Neural Network Yoshiteru Takezawa, Hidekazu Suzuki, Mamoru Minami, Yasushi Mae SICE Annual Conference 2005 in Okayama PROCEEDINGS, TP1-03-2, 2400-2405 (2005)

[4] Prediction of Fish Motion by Neural Network Y. Li, Y. Takezawa, H. Suzuki, M. Minami, Y. Mae The 3rd International Symposium on Autonomous Minirobots for Research and Edutainment (2005)

[5] M.Minami, T.Yoshida:"Prediction Servoing to Catch Escaping Fish Using Neural Network", 2008 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics Proc., pp.1225-1231, 2008

[6] R.A.Brooks:"Model-Based Three-Dimensional Interpretations of Two-Dimensional Images", IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI-5, 2, pp.140-150, 1983

[7] M.Minami, J.Agbanhan, and T.Asakura:"GA-Model-based Object Recognition Using Real-World Gray-Scale Image", IMEKO-XV, World Congress of Int. Measurement Confederation, No.17.2.4, 1999

[8] M.Minami, J.Agbanhan, and T.Asakura:"Robust Scene Recognition Using a GA and Real-world Raw-image", Measurement, 29, pp.249-267, 2001

[9] 3D Visual Servoing by Feedforward Evolutionary Recognition W.Song, Y.Fujia, M.Minami, Journal of Advanced Mechanical Design, Systems, and Manufacturing (JSME), Vol.4, No.4, pp.739-755, 2010

[10] M.Minami, H.Suzuki, Julien AGBANHAN："Fish Catching by Robot Using Gazing GA Visual Servoing", The Japan Society of Mechanical Engineers, C-68-668, pp.1198-1206, 2002

[11] Keita Mori, Yuya Ito, Mamoru Minami, Akira Yano："Relative Intelligence Evaluation of Robot and Fish by Continuous Capturing and Releasing Experiments", The 57th Annual Conference of the Institute of Systems, Cotrol and Information Engineers