ICROS-SICE International Joint Conference 2009
August 18-21, 2009, Fukuoka International Congress Center, Japan

Fish-catching by Robot Using Prediction Neural Network -Reducing Steady-state Error to Zero-

Mamoru Minami[1] and Tongxiao Zhang[2]

[1]Faculty of Engineering, University of Fukui, Bunkyo3-9-1, Fukui, Japan
(Tel,Fax: +81-776-27-8527; E-mail: minami@rc.his.u-fukui.ac.jp)
[2]Graduate school of Engineering, University of Fukui, Bunkyo3-9-1, Fukui, Japan
(Tel,Fax: +81-776-27-8527; Email: zhangtongxiao@rc.his.u-fukui.ac.jp)

Abstract: This paper presents a method to predict a fish motion by Neural Network (N.N.) with on-line learning when a robot is pursuing fish-catching by a net at hand through hand-eye robot visual servoing. We have learned by previous experiments that fish is much smarter than a robot controlled by visual servoing whose escaping strategy is to make a steady state distance error between the net at robot's hand and the fish. To overcome the fish's escaping strategy we propose prediction servoing utilizing estimated future fish position by on-line adjusting N.N.. The effectiveness have been proven through visual servoing and fish catching experiments.

Keywords: Fish-Catching, Neural Network.

1. INTRODUCTION

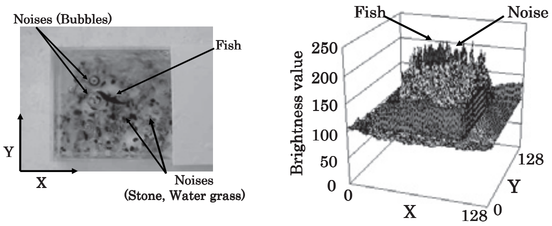
Researches on catching a fish based on object feature recognition and visual servoing has been performed [1][2][3]. But as the catching operation by a net attached at the hand continues, the fish in the pool gradually learned the action pattern of the robot hand and began to generate intelligent avoiding behaviors against the net that keeps chasing it. We can treat this kind of fish strategy as innate intelligence to avoid its predator. As a result, the system controlled under the traditional robot system such as visual servoing based on velocity PD feedback control failed to catch the smart fish occasionally. That is the fish has found new strategies to escape from the net pursuing it consistently. To overcome this intelligence of the fish to survive, we challenged to construct a more intelligent robot on the purpose of exceeding the fish's intelligence for successful tracking and catching operation.

Under the circumstance of fish's avoiding behaviors from the net, we consider that the robot can track the fish and catch it easier if the control system can predict its future motion. Thus, N.N.[4][5] is adopted to the current robot system to learn the fish motion pattern and predict the future position[6][7]. We have tried to predict the fish position in the future based on the position profile of the fish in the past. But it did not give us good prediction performance. In this paper, we propose a new visual servoing approach, named prediction servoing that can decrease the steady state error between the net and the fish error by using neural network learning the fish's future trajectory through circular approximation and resulted actual fish's trajectory.

2. REAL-TIME RECOGNITION

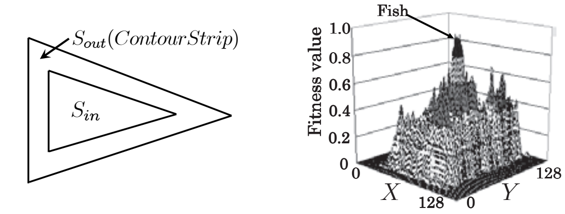
Consider the 2-D raw-image of a target fish shown in Fig.1(a), its corresponding 3-D plot is shown in Fig.1(b). In Fig.1(b), the vertical axis represents the image brightness values, and the horizontal axes represent the image plane. To search for such a target fish in the raw-image, a geometrical triangular 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_{in} and the contour-strips S_{out}, also the combination as S, the shape of S_{in} is chosen to be the similar shape to fish's one. Since the pose in two dimensional plane of surface-strips model S varies in time, it is described by $\boldsymbol{\phi}(t) = [x(t), y(t), \theta(t)]^T \in L$, designating the pose of the origin of the model S in the image space L. Then S moves in L in time and a set or the pose of the model is expressed as $S(\boldsymbol{\phi}(t))$. Then the brightness distribution of raw-image overlapping to S is expressed as $p(\boldsymbol{r}), \boldsymbol{r} \in S(\boldsymbol{\phi})$. The correlation function $F(\boldsymbol{\phi})$ of the surface-strips model with the image is given as

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



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

Fig. 1 Raw-image of swimming fish



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

Fig. 2 Surface-strips model to search a fish

PR0002/09/0000-5020 ¥400 © 2009 SICE

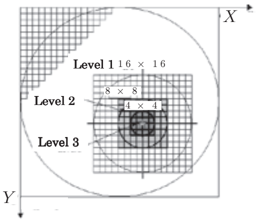


Fig. 3 Area switching

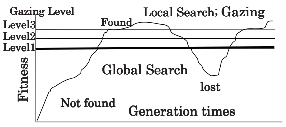


Fig. 4 Gazing switching

This means the integrated brightness difference between the internal surface and the contour-strips of the surface-strips model. The filtered result of (1) with respect to Fig.1(a) is shown in Fig.2(b). We can see the filtered result has a peak corresponding to the position of the target fish in the raw-image. This correlation using the surface-strips model means a shape-based filter since $F(\boldsymbol{\phi}(t))$ takes into account the integration and differentiation based on the object shape and the background noise simultaneously, and we think this character is effective for such noisy image as shown in Fig.1(a). Then the problem of recognition of a fish and detection of its pose is converted to an optimization problem of $F(\boldsymbol{\phi}(t))$ as

$$\boldsymbol{\phi}^{max}(t) = \left\{\boldsymbol{\phi}(t) \;\middle|\; \max_{\boldsymbol{\phi}\in L} F(\boldsymbol{\phi}(t))\right\}. \tag{2}$$

To recognize a target in a dynamic image input by video rate, 33 [fps], the recognition system must have real-time nature, that is, the searching model must converge to the fish in the successively input raw images. We adopt "1-Step GA"[3].

3. GLOBAL/LOCAL GA SEARCH

We employ an elitist preservation strategy of GA. The genes in GA possess the information $\boldsymbol{\phi}_i (i = 1, 2, \cdots, n)$ of the position and orientation of $i\text{-}th$ searching model. Every generational stage of simple GA's evolution, selection, crossover and mutation operators are performed to evolve the genes toward the true position and orientation $\boldsymbol{\phi}_p = [x_p, y_p, \theta_p]$ of the target fish in the raw-image. The global GA search process in the loop, always make efforts to find the highest peak indicating the fish in the whole camera image. When the best searching model indicated by the best gene has the highest fitness value, which means that it matches to the fish in the raw-image, the x, y, θ and x_p, y_p, θ_p get the same corresponding values.

Like you and me, if we were to track a moving target with our eyes, at a certain degree we do not look at the surrounding and just focus our attention on the target, that is gazing it. Here, we think of how to propose a similar action as the human being to real-time visual servoing. In our proposed local search method, the genes of the best individual is selected to be copied to all of the other individuals, thus making an intermediate population of identical individuals for possible solution of the subsequent population. Next in the reproduction process, except for that first ranking individual, in order to increase the fitness value of the others and at the same time to obtain better position and orientation results, a mutation operation is gradually performed on the lower level bits of the genes. When the mutation is limited to four bits of the lower level, sixteen reproduction patterns can be obtained for one positional direction. When the mutation is limited to three bits, eight patterns can be obtained and when it is limited to two bits, four reproduction patterns can be obtained. Fig.3 depicts the three level of gazing area and the level is determined by the highest fitness value, representing how much degree the possible solution matches to the position and orientation of the target fish in the raw-image. The transition of the gazing level is depending on the highest gene's fitness value as illustrated in Fig.4. As you see in Fig.4, the local search technique focuses on the highest point. Thus, faster and correct detection of the target will be possible. Thus, faster and correct detection of the target will be possible.

In practice, once the global GA has achieved the stage of the detection of a target, after reaching a certain fitness value as a threshold value, it switches to the local GA that performs the fine and fast recognition of the target. Using the combined GA, the task executed by machine to search for a target and track the fish can be thought to be similar to the same task done by human.

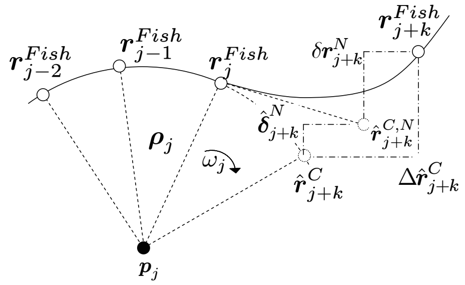

$\boldsymbol{r}^{Fish}_{j-2}$
$\boldsymbol{r}^{Fish}_{j-1}$
$\boldsymbol{r}^{Fish}_{j}$
$\delta\boldsymbol{r}^{N}_{j+k}$
$\boldsymbol{r}^{Fish}_{j+k}$
$\hat{\boldsymbol{\delta}}^{N}_{j+k}$
$\hat{\boldsymbol{r}}^{C,N}_{j+k}$
$\boldsymbol{\rho}_j$
ω_j
$\hat{\boldsymbol{r}}^{C}_{j+k}$
$\Delta\hat{\boldsymbol{r}}^{C}_{j+k}$
$\boldsymbol{p}_j$

Fig. 5 Circular approximation by using N.N.

4. NEURAL NETWORK LEARNING

4.1 Circular Approximation

Assuming a line to be a circle with an infinite radius, any three coordinates can be connected by one circle. Here, $\boldsymbol{r}^{Fish}_{j-2}$ and $\boldsymbol{r}^{Fish}_{j-1}$ denote the past fish position coordinates, and $\boldsymbol{r}^{Fish}_{j}$ does the current fish position coordinates as shown in Fig.5. Similarly $\hat{\boldsymbol{r}}^{C}_{j+k}$ denotes the predicted fish position coordinates in the future k-control period based on the circular approximation, and $\boldsymbol{p}_j = (p_j, q_j)$ does the center coordinates of approximated circle at current time t. Here we denote $\boldsymbol{\rho}_j = \boldsymbol{r}^{Fish}_{j} - \boldsymbol{p}_j$ and $\Delta\boldsymbol{\rho}_j = \boldsymbol{\rho}_j - \boldsymbol{\rho}_{j-1}$, and consider the equation shown as

$$\boldsymbol{t} = \boldsymbol{\rho}_j \times \Delta\boldsymbol{\rho}_j, \tag{3}$$

where × denotes outer product. 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 adopt-

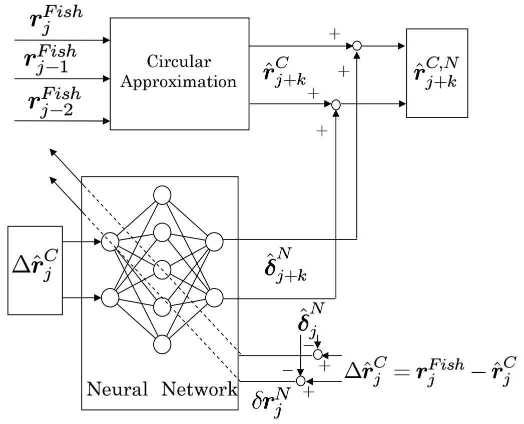


Fig. 6 Block diagram of prediction

ing radius and point $j-1$ to point j as

$$\omega_j \fallingdotseq sign(t_z)\frac{|\Delta\boldsymbol{\rho}_j|}{\Delta t \cdot r_j}, \tag{4}$$

$$r_j = |\boldsymbol{r}_j^{Fish} - \boldsymbol{p}_j|. \tag{5}$$

r_j instantaneous radius of the circular trajectory of the fish is calculated in (5). α expresses the angle of the current fish position based on the horizontal line shown in Fig.5. Then $\boldsymbol{r}_j^{Fish} = [x_j, y_j]^T$ is

$$\boldsymbol{r}_j^{Fish} = \boldsymbol{p}_j + \begin{bmatrix} r_j \cos\alpha \\ r_j \sin\alpha \end{bmatrix}. \tag{6}$$

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

$$\hat{\boldsymbol{r}}_{j+k}^C = \boldsymbol{p}_j + \begin{bmatrix} r_j \cos(\alpha + k\omega_j\Delta t) \\ r_j \sin(\alpha + k\omega_j\Delta t) \end{bmatrix}. \tag{7}$$

4.2 N.N.-Learning by Circular Approximation Error

Should predicted future fish posituin $\hat{\boldsymbol{r}}_{j+k}^C$ and actual fish position $\boldsymbol{r}_{j+k}^{Fish}$ at time $(j+k)\Delta t$ be exactly same, the circular approximation error is $\mathbf{0}$. However since it is very rare that the fish's trajectory should coincide with the circular orbit, the error $\Delta\hat{\boldsymbol{r}}_{j+k}^C$ appears in the future time at $(j+k)\Delta t$ as

$$\Delta\hat{\boldsymbol{r}}_{j+k}^C = \boldsymbol{r}_{j+k}^{Fish} - \hat{\boldsymbol{r}}_{j+k}^C. \tag{8}$$

Here the definition of prediction error E_{j+k} is expressed at $k\Delta t[s]$ in the future as

$$E_{j+k} = |\Delta\hat{\boldsymbol{r}}_{j+k}^C|. \tag{9}$$

The block diagram of future fish position prediction using N.N. being learned by Back Propagation (B.P.) is shown Fig.6. First $\hat{\boldsymbol{r}}_{j+k}^C$ is calculated by circular approximation using (7). The input of the N.N. at $j\Delta t = t$, $\Delta\hat{\boldsymbol{r}}_j^C$, is the error vector between current fish's position $\boldsymbol{r}_j^{Fish}$ and predicted position errors $\hat{\boldsymbol{r}}_j^C$ at $(j-k)\Delta t$, which ascends back $k\Delta t[s]$ from current time,

$$\Delta\hat{\boldsymbol{r}}_j^C = \boldsymbol{r}_j^{Fish} - \hat{\boldsymbol{r}}_j^C. \tag{10}$$

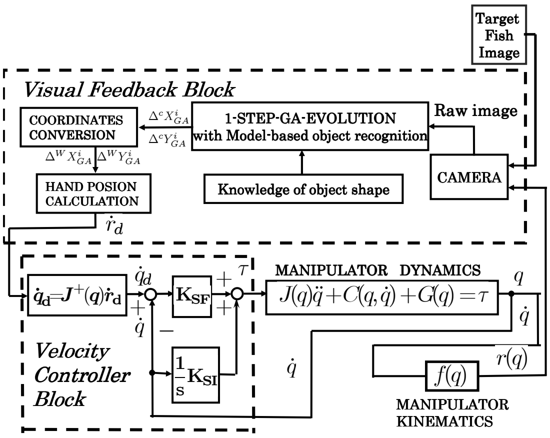


Fig. 7 Block diagram of the controller

As being shown in Fig.5, the predicted output of N.N. at future time $(j+k)\Delta t$ is represented by $\hat{\boldsymbol{\delta}}_{j+k}^N$. Then the predicted position $\hat{\boldsymbol{r}}_{j+k}^{C,N} = (\hat{x}_{j+k}^{C,N}, \hat{y}_{j+k}^{C,N})$ is calculated by adding N.N. output $\hat{\boldsymbol{\delta}}_{j+k}^N$ predicted at $j\Delta t$ to circular approximation restlt $\hat{\boldsymbol{r}}_{j+k}^C$,

$$\hat{\boldsymbol{r}}_{j+k}^{C,N} = \hat{\boldsymbol{r}}_{j+k}^C + \hat{\boldsymbol{\delta}}_{j+k}^N. \tag{11}$$

In the block diagram of Fig.6, final predicted position obtained through summation of circular approximation and correction of N.N., that is $\hat{\boldsymbol{r}}_{j+k}^{C,N}$ is calculated by (11).

The teaching signal of N.N., $\delta\boldsymbol{r}_j^N$ is

$$\delta\boldsymbol{r}_j^N = \Delta\hat{\boldsymbol{r}}_j^C - \hat{\boldsymbol{\delta}}_j^N. \tag{12}$$

Since N.N. always learns to decrease the error $\delta\boldsymbol{r}_j^N$ to $\mathbf{0}$ by B.P., N.N. changes its coefficients for such objective as

$$\delta\boldsymbol{r}_j^N \to \mathbf{0} \quad (t \to \infty). \tag{13}$$

Therefore,

$$\Delta\hat{\boldsymbol{r}}_j^C = \hat{\boldsymbol{\delta}}_j^N \tag{14}$$

is derived from (12). Then considering (8), (11) and (14), we get

$$\hat{\boldsymbol{r}}_j^{C,N} \to \boldsymbol{r}_j^{Fish} \quad (t \to \infty), \tag{15}$$

and this describes circular approximated position with N.N. compensation at $j\Delta t$ will converge into the actual fish position $\boldsymbol{r}_j^{Fish}$.

5. FISH TRACKING/CATCHING CONTROL

5.1 Visual Servoing

The experimental system is explained as follows. The camera-to-fish distance is 450 [mm]. The size of the water pool is 390×460×100 (depth) [mm], and the net is 100×125 [mm]. Catching the fish is executed by pulling up the net when the fish is within an area of 60×80 [mm]

Table 1 Gain parameters

$\boldsymbol{K}_P$	[0.95 0.95]
$\boldsymbol{K}_V$	[0.60 0.60]
Link	[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]

at the center of the net. The range area of the camera view is 150 and 120[mm] in x and y directions. We have shown in section 2 that $\boldsymbol{r}_n^{Fish}$ represents the current fish position by "1-Step GA". In the 2-D servoing experiment, which fixes the camera-to-fish distance, the camera calibration using a parameter such as camera focal length is not performed.

The aforementioned real-time recognition system in section 2 using the shape of the fish as the knowledge base is depicted in the upper side of the block diagram in Fig.7. In the figure, $\Delta\boldsymbol{r} = [\Delta^W X_{GA}^i, ^W Y_{GA}^i]$ is the X-Y deviation from the camera center to the fish expressed in the world coordinates.

To make the based coordinates Σ_C, the camera coordinates, to be clear, superscript "C" is added as ${}^C\boldsymbol{r}_n^{Fish}$. The error vector of the net position to the fish, $\Delta^C\boldsymbol{r}_n$ is

$$\Delta^C\boldsymbol{r}_n =^C \boldsymbol{r}_n^{Fish} -^C \boldsymbol{r}_n^{Net}. \tag{16}$$

Here the net position in Σ_C is constant and set to be ${}^C\boldsymbol{r}_n^{Net} = \boldsymbol{0}$ by having the origin of Σ_C to be the center of the net. $\Delta^C\boldsymbol{r}_n$ is converted to the position $\Delta^W\boldsymbol{r}_n$ described in Σ_C through 3×3 matrix of ${}^W\boldsymbol{R}_C$ expressing the orientation relation of Σ_W and Σ_C as

$$\Delta^W\boldsymbol{r}_n = {}^W\boldsymbol{R}_C \Delta^C\boldsymbol{r}_n. \tag{17}$$

Using the error $\Delta^W\boldsymbol{r}_n$, the desired hand velocity is given as

$$\dot{\boldsymbol{r}}_n^d = \boldsymbol{K}_P\Delta^W\boldsymbol{r}_n + \boldsymbol{K}_V(\Delta^W\boldsymbol{r}_n - \Delta^W\boldsymbol{r}_{n-1}), \tag{18}$$

where $\boldsymbol{K}_P$ and $\boldsymbol{K}_V$ are matrices of positive definite.

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

$$\dot{\boldsymbol{q}}_d = \boldsymbol{J}^+(\boldsymbol{q})\dot{\boldsymbol{r}}_d\ , \tag{19}$$

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

$$\boldsymbol{\tau} = \boldsymbol{K}_{SP}(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}) + \boldsymbol{K}_{SI}\int_0^t (\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}})dt\ , \tag{20}$$

where $\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}$ is the velocity error of the joint angle, $\boldsymbol{K}_{SP}$ and $\boldsymbol{K}_{SI}$ are symmetric positive definite matrices to determine PI gain (Table1). The orientation of the fish is measured in real time, but in the tracking and catching experiment, the measured orientation information is not considered as shown in the above equation. The manipulator servo update rate is 100[Hz]. A diagram describing the experimental set up is shown in Fig.8.

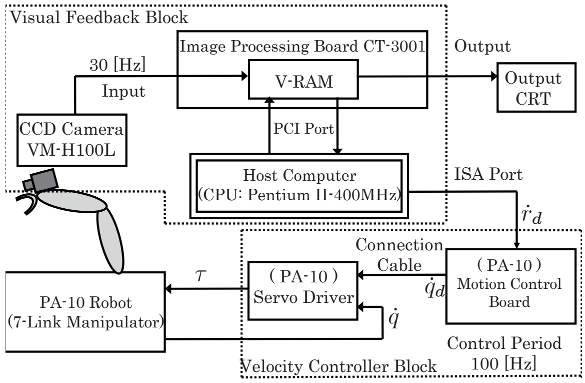


Fig. 8 Experimental set up

5.2 Prediction Servoing

As we know the nature of PD velocity servo system that the system driven by equations from (18) to (20) suffers steady state position error of $\Delta^W\boldsymbol{r}_n$ when the fish swims with constant speed, this servo system cannot catch the fish effectively, which will be shown later by experiments. To overcome this defect of PD servo, we devised a new velocity servo controller whose control error is calculated by the distance error between predicted fish position in future time $\hat{\boldsymbol{r}}_{n+k}^{C,N}$ and current fish position. We named this servoing method as "Prediction Servoing". By replacing $\Delta^W\boldsymbol{r}_n$ in (18) into $\hat{\boldsymbol{r}}_{n+k}^{C,N}$ defined as

$$\Delta^W\hat{\boldsymbol{r}}_{n+k}^{C,N} = {}^W\boldsymbol{R}_C\ \Delta^C\hat{\boldsymbol{r}}_{n+k}^{C,N}, \tag{21}$$

$$\Delta^C\hat{\boldsymbol{r}}_{n+k}^{C,N} = {}^C\hat{\boldsymbol{r}}_{n+k}^{C,N} -^C \boldsymbol{r}_n^{Net}, \tag{22}$$

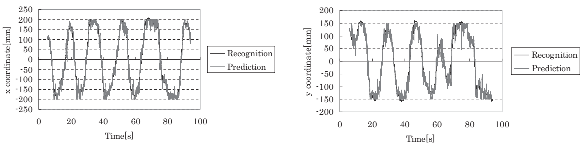
we define the prediction controller as follows,

$$\dot{\boldsymbol{r}}_n^d = \boldsymbol{K}_P\Delta^W\hat{\boldsymbol{r}}_{n+k}^{C,N} + \boldsymbol{K}_V(\Delta^W\boldsymbol{r}_n - \Delta^W\boldsymbol{r}_{n-1}). \tag{23}$$

It will be noticed that $\Delta^W\boldsymbol{r}_n$ is used in (23) for velocity feedback. 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 $\Delta^W\hat{\boldsymbol{r}}_{n+k}^{C,N}$. $\boldsymbol{K}_P, \boldsymbol{K}_V, \boldsymbol{K}_{SP}, \boldsymbol{K}_{SI}$ are shown in Table 1.

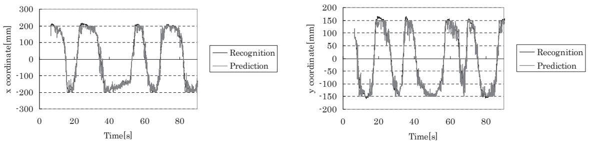
6. EXPERIMENTS

Here we will evaluate the effectiveness of simple visual servoing control and the proposed prediction servoing control by some experiments. The target fish is only one and the tracking time is 90[s] continuously. From the detected position in real time, the future position after $k \cdot \Delta t = 0.48$[s] from current time is predicted, and the predicted one is compared with the actual fish position at the same time after 0.48[s] has passed. During this tracking experiment a net attached at the hand is moving with the hand motion at the bottom of the pool in horizontal



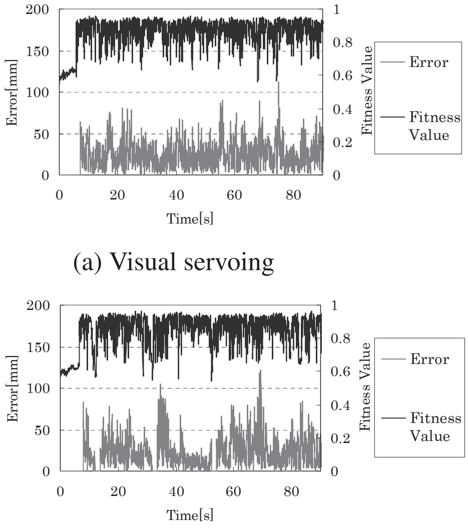
(a) x position (b) y position

Fig. 9 Fish position during visual servoing



(a) x position (b) y position

Fig. 10 Fish Position during prediction servoing



(a) Visual servoing

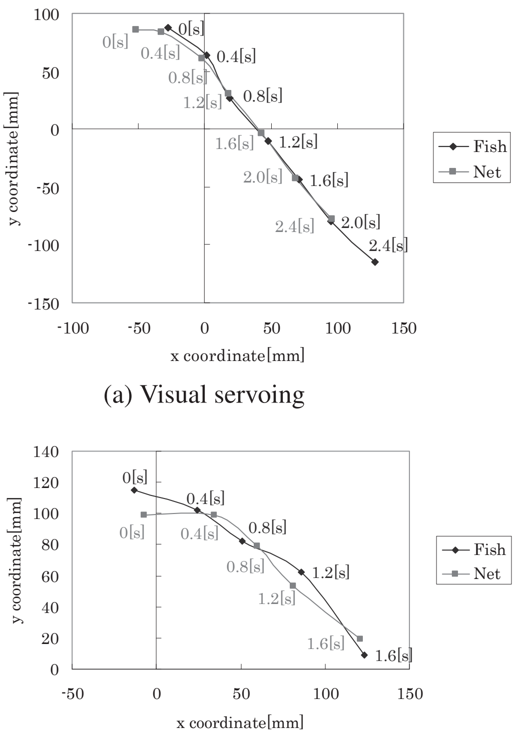
(b) Prediction servoing

Fig. 11 Recognition and error results

x-y plain. This means the net always give the fish threat but it does not execute a picking operation.

Fig.9 shows x and y trajectories of actual position $\boldsymbol{r}_n^{Fish}$ and predicted one $\hat{\boldsymbol{r}}_{n+k}^{C,N}$ under visual servoing control during 90[s]. Both trajectories are almost overlapping, meaning the visual servoing works stably and prediction task also. During the experiment fitness value of 1-step GA and the distance error between fish and net position $|\Delta^C \boldsymbol{r}_n|$ is shown in Fig.11(a). Since the fitness value is almost to be 1.0, representing the recognition worked stably. The steady state error of $|\Delta^C \boldsymbol{r}_n|$ is about 40[mm]. Fig.10 is x and y trajectories of actual position and predicted one under the prediction control. The prediction servo control is stable and during the control real-time recognition of the fish is confirmed to be successfully stable. Fig.11(b) is on the condition of prediction servoing. The difference from Fig.11(a) is that the position error changes gradually, meaning $|\Delta^C \boldsymbol{r}_n|$ sometimes decreases to about 10[mm] and this value is small enough to catch the fish by pulling up the net.

Further data of visual servoing are shown in Figs.12(a), 13, 15. Fig.12(a) depicts fish and net trajectories with time history, corresponding photos are in Fig.13, and the servo errors of $\Delta^C \boldsymbol{r}_n = [\Delta^C x_n, \Delta^C y_n]$ are shown in Fig.15. From these figures the PD visual servoing is not effective to catch fish because there exists steady state error, which is understandable nature of PD velocity control under the fish escaping with constant velocity.



(a) Visual servoing

(b) Prediction servoing

Fig. 12 Fish and net trajectories during visual servoing and prediction servoing

Contrarily with these results prediction servoing performances shown in Figs.12(b), 14 and 16 represent the steady state error decreases and it oscillates gradually around zero, indicating prediction servo may overcome the fish's strategy to escape from the PD control visual servo by generating the system steady state error. This will be confirmed in the next experiment.

About fish catching experiment, by prediction servoing, the photos taken by hand eye camera is shown in Fig.17. Net and fish trajectory with time history are shown in Fig.18. In Fig.18, the time at 0.96[s] indicates when the servoing control strategy is changed from visual servoing to prediction servoing. Before that time point "B" the net looks to be delayed with steady state error, but after that point the net rapidly catch up with the fish. The fish velocity and hand velocity are compared in Fig.19, showing at point "B" (this "B" is corresponding to the B in Fig.18) the net moves faster than the fish to accelerate and catch up. At point "C" the net velocity rapidly decreased to zero and catching operation has been triggered by recognizing the fish stays at almost the center of the net, resulting in successful catching the fish.

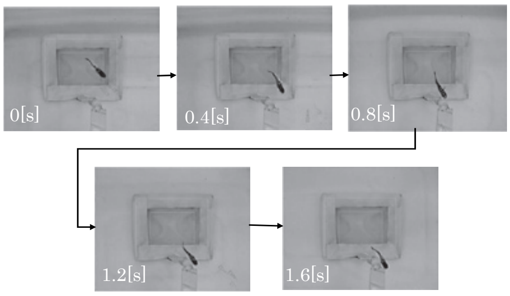


Fig. 13 Fish and net motion image in visual servoing

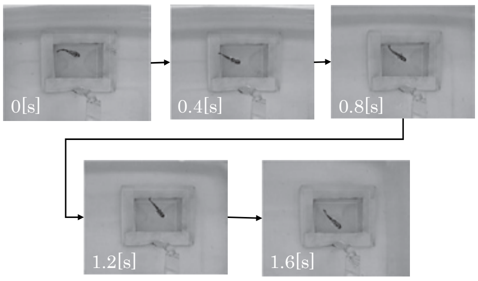
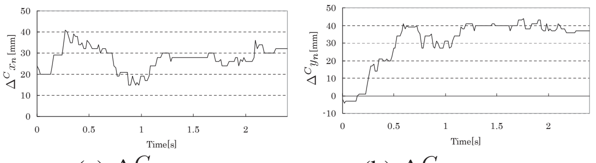
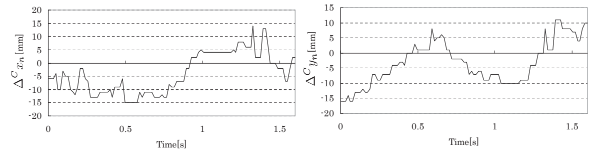


Fig. 14 Fish and net motion image in prediction servoing



(a) $\Delta^C x_n$ (b) $\Delta^C y_n$

Fig. 15 Fish position error in visual servoing



(a) $\Delta^C x_n$ (b) $\Delta^C y_n$

Fig. 16 Fish position error in prediction servoing

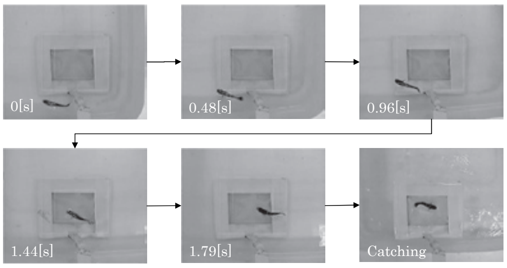


Fig. 17 Fish and net motion image in catching the fish

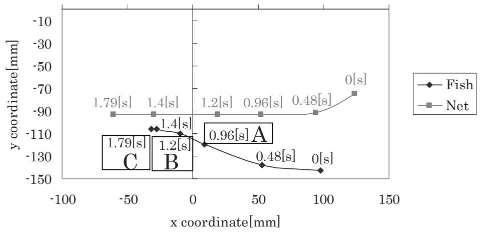


Fig. 18 Fish and net trajectory in catching the fish

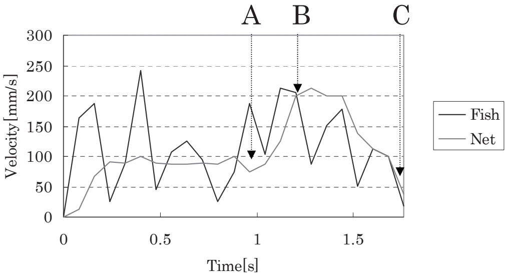


Fig. 19 Fish and net velocities

7. CONCLUSION

This paper presents a method to predict a fish motion by Neural Network (N.N.) with on-line learning when a robot is pursuing fish-catching by a net at hand through hand-eye robot visual servoing. By visual servoing experiments and prediction servoing experiments, we have shown prediction servoing can reduce steady-state error to zero and catch a escaping fish by catching up with the hand net.

REFERENCES

[1] Hiroshi Nakamine, Li Nie, Nobuo Sannomiya, “An Approximation of Fish School From by an Ellipse”, The Institute of Systems, Control and Information Engineers(ISCIE) Vol.6, No.7, pp.297-304, 1993.

[2] Hidekazu Suzuki, Mamoru Minami, “Fish Catching by Visual Servoing and Observed Intelligence of the fish“, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 287-292, 2002.

[3] Hidekazu Suzuki, Mamoru Minami, “Visual Servoing to catch fish Using Global/local GA Search“, IEEE/ASME Transactions on Mechatronics, Vol.10, Issue 3, 352-357, 2005.

[4] Hyun-Take Cho, Seul Jung, “Neural Network Position Tracking Control of an Inverted Pendulum by an X-Y Table Robot“, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1210-1215, 2003.

[5] Samy F.M. Assal, Keigo Watanabe, Kiyotaka Izumi, “Neural Network Learning from Hint for the Inverse Kinematics Problem of Redundant Arm Subject to Joint Limits“, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 821-826, 2005.

[6] Y.Takezawa,H.Suzuki,M.Minami,Y.Mae, “Learning of Fish Movement Pattern by Neural Network“SICE Annual Conference,TP1-03-2, 2005.

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