PURSUIT MOVEMENT OF PAN-TILT CAMERA BY FEEDBACK-ERROR-LEARNING

Shotaro Akaho†*and Yoshiyasu Goto*††*and Takio Kurita*†*and Hiroshi Mizoguchi*††
Email:akaho@etl.go.jp

†Information Science Division, Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba 305-8568, Japan
††Department of Information and Computer Science, Saitama University
255 Shimo-Okubo, Urawa 338-0825, Japan

ABSTRACT

We investigate the ability of feedback-error-learning (FEL) algorithm through applying it to training a pan-tilt camera head for pursuing a moving target. Although the task of pursuit eye movement is reflective and the control has a relatively large delay, our experiments show that the camera head can successfully acquire the skill of pursuit eye movement. Our experimental results also show that the performance of the learning highly depends on the gain parameter of the feedback controller module. The system sometimes oscillates or breaks down when we use the optimal feedback controller (large gain in general), because of over-training-like phenomena. If the gain is too small, the performance is not improved by learning, though the system is stable. Those results suggest the large possibility of FEL algorithm as well as the importance of the design of feedback module.

KEYWORDS:
neural control and robotics, feedback-error-learning, pursuit eye movement, biomimetic vision system

1. INTRODUCTION

Animals realize both global view and high resolution by moving their eyes appropriately. Such skills are acquired through lots of gazes to moving targets in the real world. The aim of this research is to reproduce such an adaptability of animals on artificial machines and to find a general principle from a view point of computational neuro science. Such a biomimetic approach to vision system has attracted much attention recently[1], especially because of the development of hardware (computers and pan-tilt camera heads), and also because of the increase of knowledge on the visual nerve systems of animals[2, 3, 4]. In this paper, we provide some experimental results on the learning of tracking a simple moving target by a pan-tilt camera head.

Katayama and Kawato[5] has proposed the feedback-error-learning (FEL) algorithm as a model of motor learning in a cerebellum. They say that the 'inverse dynamics' calculates the motor commands from a desired motion of hands, legs, eyes and so on. At the beginning, the FEL algorithm was intended to explain the feedforward model that controls a plant along to a pre-planned orbit, and the validity of the algorithm has shown by the task of reaching movement of robot hands. However, the algorithm can be considered in more general framework, and Gomi et al[6] extended the algorithm to the learning of reflective motion which is mainly controlled by feedback controller. Such an extension is considered to be so-called 'horizontal architecture'. There are two types of extension called the inverse dynamics model learning and the nonlinear regulator learning. In this paper, we use the nonlinear regulator learning because it was a little superior to the inverse dynamics model learning in our preliminary experiments.

We applied the FEL algorithm to the task of pursuing a moving target (secs. 2 and 3). This task is reflective in the sense that the optimal orbit can not be determined a priori, because of the randomness of the target motion. From a hardware restriction of the camera head, a relatively large delay until the control command is sent (such a delay also exists in animal vision), and also the control system includes some nonlinearity. One thing that makes the task easy is that the freedom of control of our current system does not include redundancy.

Under the above situation, the camera head could successfully acquire the skill of pursuit eye movement by FEL (sec.4.2), and we also found through the experiments that the performance of FEL highly depends on the performance of feedback controller module (sec.4.3). If the gain is too large, feedback controller itself oscillates. And under some value, although the feedback controller works very well, the system comes to oscillate as the learning is going. On the other hand, if the gain is too small, the performance is not improved by learning though the system is stable.

2. FEEDBACK-ERROR-LEARNING ALGORITHM

In this section, we summarize the feedback-error-learning (FEL) algorithm. In the framework of FEL, the system receives a desired orbit of a control target (plant) and an observed orbit of the plant. One difficulty in this framework than usual learning scheme is that the system receives only a desired status as a teacher signal, not a desired output. This is a kind of reinforcement learning scheme in a wide sense.

The system for the FEL consists of two modules: one is a feedforward controller with modifiable parameters $\boldsymbol{w}$, and the other is a feedback controller which provides rough and slow control commands to the target. As a whole, the system outputs the sum of outputs of the both controllers. If the output approximates the desired output well, the learning equation can be composed so that the output of the feedback controller is dealt with as an error signal in the steepest descent method.

Kawato has proved the FEL algorithm is considered as a Newton method in a wide sense under several conditions. However, those conditions are not satisfied completely in our task. For example, the controller has a relatively large delay, and the desired orbit is incomplete as shown later.

The block diagram of the nonlinear regulator learning model, which is proposed by Gomi as an extension of FEL, is shown in fig.1. More specification about each controllers used in our experiments are provided in the next section.

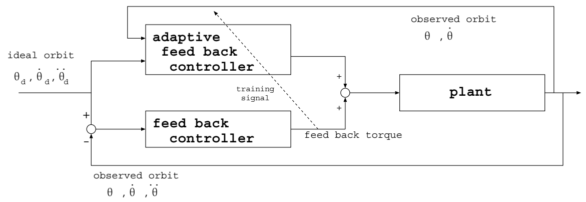

Figure 1: The block diagram of the nonlinear regulator learning model

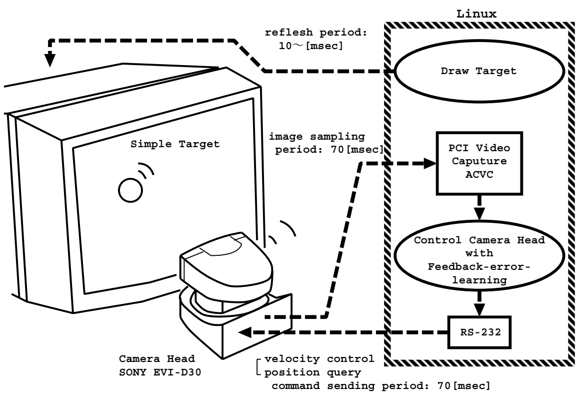

Figure 2: System architecture

3. SYSTEM SPECIFICATION

3.1. Hardware Specification

The camera head tracks a black spot displayed on CRT. The black spot is easy to recognize, and it is possible to generate various kinds of target motion by displaying it on CRT. The system architecture is shown in fig.2. The platform of the control is PC(Pentium II 200MHz) on which Linux OS is running.

The pan-tilt camera head is EVI-D30 developed by SONY, which accepts many kinds of control/inquiry commands from PC through a serial port. We can control a speed of the camera and get a current direction in about 70[msec] cycle. However, the speed control has relatively large delay, which is about 300[msec] at most until the actual speed becomes the desired one. In the experiments, we control only pan (horizontal direction), and the range of the angle is narrower than between -30[deg] and $+30$[deg].

Image data of the camera is transferred via NTSC and captured by using PCI video capture card ACVC developed by Argo Craft. In order to synchronize to the camera control, images are captured in every 70[msec] approximately.

3.2. Learning Model Specification

Inverse model We use three-layered neural network model with linear output units and direct connections between input units and output units as adaptive feedback controller of fig.1. The number of hidden nodes is fixed to 5.

Output of the adaptive feedback controller is written as

$$\tau_{\mathrm{ff}}(t) = f(\boldsymbol{w}, \theta(t), \theta'(t), \theta''(t), \theta_{\mathrm{d}}(t), \theta'_{\mathrm{d}}(t), \theta''_{\mathrm{d}}(t)), \quad (1)$$

where θ denotes the actual orbit of the camera head, and θ_{d} denotes the desired orbit. The derivatives $\theta', \theta'', \theta'_{\mathrm{d}}, \theta''_{\mathrm{d}}$ are estimated by smoothed difference of θ and θ_{d}.

Since it is difficult to obtain $\theta_{\mathrm{d}}(t)$ without any knowledge about target motion because of the randomness of the target motion, we use the following rough approximation of θ_{d},

$$\theta_{\mathrm{d}}(t) = \theta(t) + kx(t) \quad (2)$$

where $x(t)$[pixels] is a distance between the target and the center of fovea, and k is a prefixed constant.

Feedback controller Feedback controller with fixed parameters are designed as the sum of PDA errors weighted by gain parameters:

$$\tau_{\mathrm{fb}} = K_{\mathrm{p}}(\theta_{\mathrm{d}} - \theta) + K_{\mathrm{d}}(\theta'_{\mathrm{d}} - \theta') + K_{\mathrm{a}}(\theta''_{\mathrm{d}} - \theta''). \quad (3)$$

Total output The system outputs the sum of the outputs of the inverse model and the feedback controller as the speed command, $\tau_{\mathrm{tot}} = \tau_{\mathrm{ff}} + \tau_{\mathrm{fb}}$.

FEL algorithm The learning equation of FEL is given by

$$\delta \boldsymbol{w} = \alpha \tau_{\mathrm{fb}} \frac{\partial \tau_{\mathrm{ff}}}{\partial \boldsymbol{w}}, \quad (4)$$

where α is a learning constant and we took $\alpha = 10^{-4}$ in the experiments. The above equation approximates the error signal $\tau_{\mathrm{tot}} - \tau_{\mathrm{d}}$ by τ_{fb}.

4. EXPERIMENTAL RESULTS

4.1. Target motion

We use time variant triangular functions as target motion. Let us introduce the deviation of the period and the amplitude of the target, T_v and A_v. The target changes its period in the uniform distribution between $[T_0(1-T_v), T_0(1+T_v)]$, where T_0 is a prefixed constant. The amplitude is similarly changes with respect to the uniform distribution $[A_0(1-A_v), A_0(1+A_v)]$. In order to avoid the discontinuity and to keep the target around the origin, we connect the variable triangular function smoothly. An example of target motion is given in fig.3.

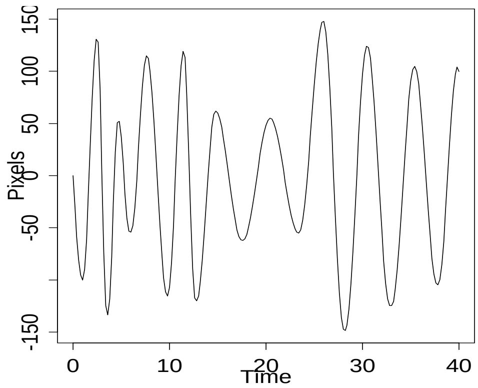

Figure 3: An example of target motion ($T_v = 0.5, A_v = 0.5$)

We tried the following variations: (1) Fast, fixed period ($T_0 = 2$[sec], $T_v = 0$), (2) Slow, fixed period ($T_0 = 6$[sec], $T_v = 0$), and (3) Varying period between fast and slow ($T_0 = 4$[sec], $T_v = 0.5$). For each target, we took $A_v = 0$ and $A_v = 0.5$, also experiments were made for different gain parameter K_p (K_d and K_a were fixed).

4.2. Learning Curve

An example of the error curve controlled by only feedback module is shown in fig.4, where the target varies its period, and MSE (mean square error) is calculated on the moving window of size 4[sec]. The performance of FEL for the target with the same parameter is given in fig.5. The effect of learning looks apparent in this figure. However, the performance depends on the value of K_p as shown in the next section.

4.3. Sensitivity to Gain Parameter

The performance of feedback controller itself depends on K_p as shown in fig.6, where MSE is calculated over some time intervals (1500–2500 in fig.6). As K_p increases, the MSE for most targets decreases drastically, and the performance suddenly break down at a critical point, which is about $K_\mathrm{p} = 10.0$ in the experiments and almost independent of the target motion.

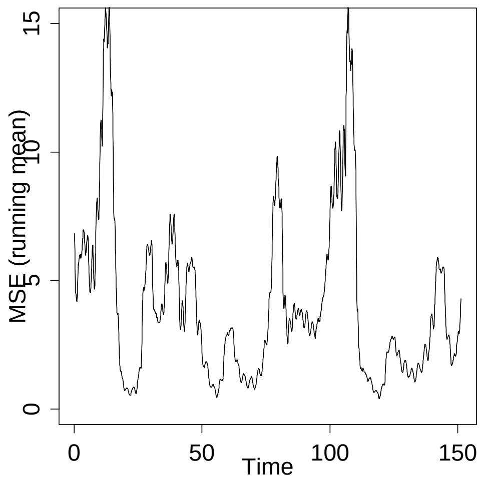

Figure 4: Moving average of MSE for feedback only control. ($T_0 = 4$[sec], $T_v = 0.5$, $A_v = 0.5$, $K_\mathrm{p} = 7.0$)

Figs. 7 and 8 are the ratio of MSE between feedback-only and FEL, where each figure shows the performance in the different time intervals. Plotted values are ratio$(K_\mathrm{p}) \equiv (\min_{K'_\mathrm{p}} R_\mathrm{fb}(K'_\mathrm{p}))/R_\mathrm{ff}(K_\mathrm{p})$, where $R_\mathrm{ff}(K_\mathrm{p})$ is MSE of FEL, and $R_\mathrm{fb}(K_\mathrm{p})$ is MSE given by fig.6. If the value is less than 1.0, the performance of FEL is inferior to the optimal feedback controller.

If K_p is very small (less than $K_\mathrm{p} \simeq 1.0$), the FEL takes a lot of time to be superior to feedback error or sometimes it is getting worse. One reason is considered to be that the feedback signal is not an approximation of error signal at all.

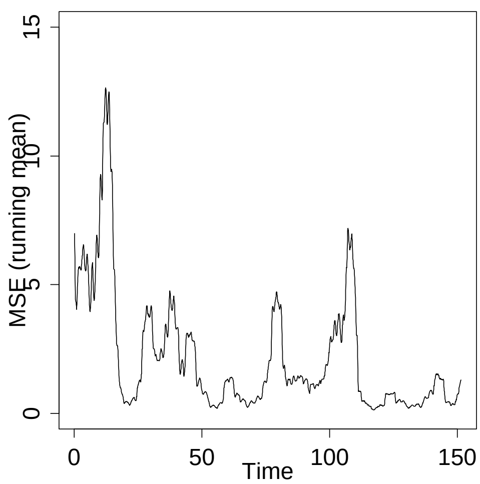

Figure 5: Moving average of MSE for FEL.($T_0 = 4$[sec], $T_v = 0.5$, $A_v = 0.5$, $K_\mathrm{p} = 7.0$)

On the other hand, in a large range of K_{p}, as time is going, the breakdown occurs at the smaller K_{p} value than the critical value in fig.6. Therefore, even if the performance of feedback controller is optimal, the FEL algorithm may cause the breakdown of the system.

As a conclusion, we should choose the feedback controller which works modestly (but not too modestly) and the system should stop learning before breakdown by observing some diagnostic value.

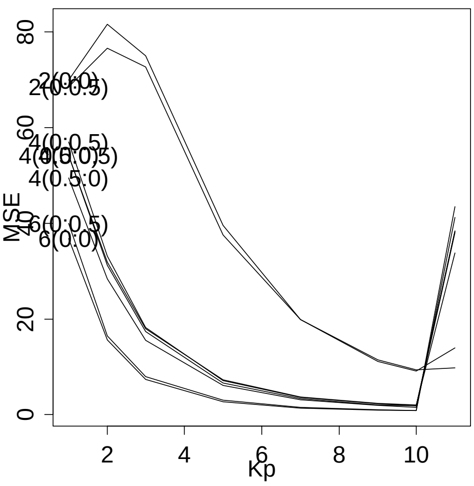

Figure 6: Gain dependence (MSE of Feedback control R_{fb}). Time steps 1500–2500. The numbers along the graph indicates $T_0(T_v : A_v)$

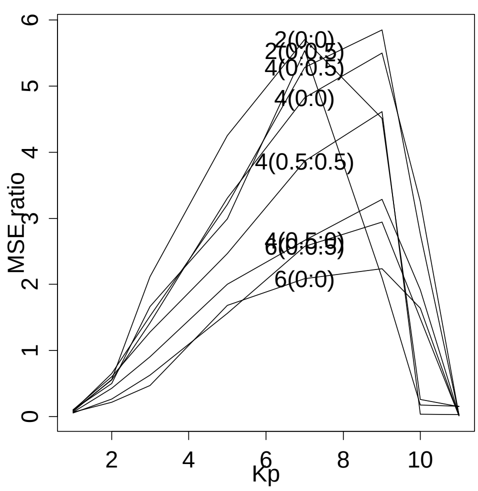

Figure 7: MSE ratio ($\min_{K'_{\mathrm{p}}} R_{\mathrm{fb}}(K'_{\mathrm{p}})/R_{\mathrm{ff}}(K_{\mathrm{p}})$). Time steps 4000–5000. The numbers along the graph indicates $T_0(T_v : A_v)$

5. CONCLUSION

We investigated the performance of feedback-error-learning algorithm through the pursuit tracking task.

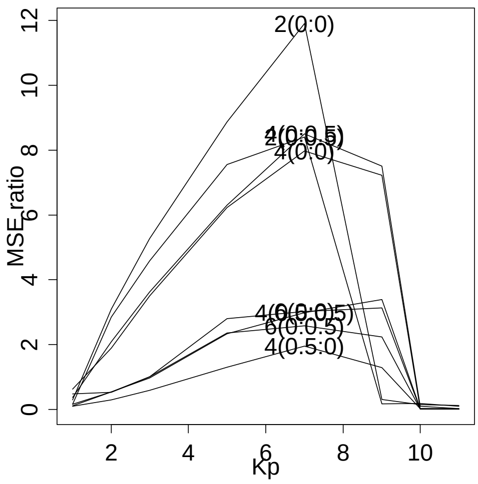

Figure 8: MSE ratio ($R_{\mathrm{ff}}(K_{\mathrm{p}})/\min_{K_{\mathrm{p}}} R_{\mathrm{fb}}(K_{\mathrm{p}})$). Time steps 9000–10000. The numbers along the graph indicates $T_0(T_v : A_v)$

Especially the gain dependency of the algorithm is considered. There is a trade-off between the performance and the stability, which reminds us of the over-training in statistical learning.

The current research is preliminary work in our research on biomimetic vision. The system will be extended to more biomimetic, including not only binocular vision and more freedom of control, but also the synthesis with learning in early vision system.

References

[1] Kuniyoshi, Y., Kita, N., Rougeaux, S. and Suehiro, T.: Active stereo vision system with foveated wide angle lenses, *2nd Asian Conference on Computer Vision*, Singapore, I:359–363(1995).

[2] Ito, M., Sakurai, M., Tongroach, P.: Climbing fibre induced depression of both mossy fibre responsiveness and glutamate sensitivity of cerebellar Purkinje cells. *Journals of Physiology London*, **324**, 113–134(1982).

[3] Miall, R.C., Wolpert, D.M. : Forward Models for Physiological Motor Control. *Neural Networks*, **9**, 1265–1279 (1996).

[4] Sakurai, M.: Synaptic modification of parallel fiber-Purkinje cell transmission in *in vivo* guinea pig cerebellar slices. *Journal of Physiology London*, **394**, 463–480(1987).

[5] Katayama, M., Kawato, M.: Learning trajectory and force control of an artificial muscle arm by parallel-hierarchical neural network model. In Lippman, R.P., Moody, J.E., Touretzky, D.S., (Eds.), *Advances in neural information processing Systems* , **3**, San Mateo,CA:Morgan Kaufmann Publishers, 436–442(1991).

[6] Gomi, H., Kawato, M.: Recognition of manipulated objects by motor learning with modular architecture networks. *Neural Networks*, **6**, 485–497 (1993).