APPLICATION OF DIRECT-VISION-BASED REINFORCEMENT LEARNING TO A REAL MOBILE ROBOT

Masaru IIDA, Masanori SUGISAKA and Katsunari SHIBATA

Dept. of Electr. & Electronic Eng.,Oita Univ., 870-1192, Japan
iida,shibata@cc.oita-u.ac.jp

ABSTRACT

In this paper, it was confirmed that a real mobile robot with a simple visual sensor could learn appropriate actions to reach a target by Direct-Vision-Based reinforcement learning (RL). The learning was performed on-line without any advance knowledge and any helps of humans. In Direct-Vision-Based RL, row visual sensory signals are put into a layered neural network directly, and the neural network is trained by Back Propagation using the training signal that is generated based on reinforcement learning. By a character of the visual sensor, when the target is located on the right of and just in front of the robot, the robot can not distinguish the target object from the background. However, the robot could obtain the actions to avoid such states and to reach the target.

1. INTRODUCTION

Reinforcement learning is an attractive method as an autonomous learning for autonomous robots, and is utilized to obtain the appropriate mapping from state space to action space as shown in Fig. 1. By combining reinforcement learning and a neural network, continuous states and actions can be dealt with because non-linear functions with continuous input and output values can be approximated by the neural network. This combination has been applied to non-linear control tasks[1][2] and games[3].

Among many kinds of sensors for a robot, a visual sensor has a lot of sensory cells, and gives huge pieces of information about the environment to the robot. Our human also depends deeply on the visual information to know the environment state. Asada et al. applied reinforcement learning to real soccer robots with a visual sensor, and proposed an autonomous state space construction method [4]. The state space is constructed by dividing the hyper space with the axises of the ball size, ball location, average goal height and so forth on the visual image. In order to decide such axises, sufficient knowledge about the given task and environment is required. The authors believe that it is important for realizing intelligence in robots that such basical knowledge should be acquired or flexibly modified by itself.

Based on this idea, one of the authors proposed Direct-Vision-Based reinforcement learning (RL), in which raw sensory signals are put into a layered neural network directly, and the network is trained by the supervised signal generated based on reinforcement learning. The effectiveness of Direct-Vision-Based RL has been confirmed only on some simulations. In this paper, it is shown that a real mobile robot with a monochrome visual sensor can learn appropriate motions from scratch without any advance knowledge in "going to a target task".

2. DIRECT-VISION-BASED REINFORCEMENT LEARNING

As described above, sensory signals are put into a neural network directly in Direct-Vision-Based RL. By this, it can be expected that whole the process from sensors to motors including recognition, attention, memory, control, conversation and so on can be emerged autonomously, adaptively, purposively and in harmony in the neural network without being divided into some function modules as shown in Fig. 2. That is different from the conventional reinforcement learning that is only for action planning, in other words, control in a wide meaning as shown in Fig. 1. Of course, there is no doubt that more complicated and recurrent-type of neural network is required. The authors believe that this approach must be essential to realize the robot intelligence like humans' even if it seems meaningless and wasteful at a glance.

Each sensory signal is represent local information. As well as RBF(Radial Basis Function)-based network and CMAC, localizing the global information makes the learning of strong non-linear function fast and stable[7]. Actually, in "going to a target" task, the learning is faster and more stable when the inputs are visual sensory signals than when two dimensional relative location of the target object are the inputs [6].

This research was supported by (1)the Grants-in-Aid for scientific Research of the Ministry of Education, Culture, Sports, Science and Technology of Japan (#13780295), and (2)Plan and Coordination Council of Exchange among Industry, Academy and Government in Oita.

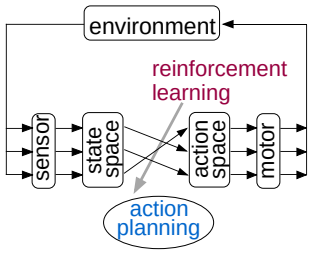

Figure 1: Conventional reinforcement learning.

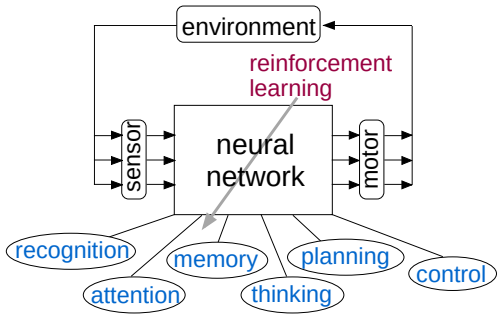

Figure 2: Direct-Vision-Based Reinforcement Learning.

The hidden neurons can represent global information by integrating the local signals adaptively[8]. In the "going to a target" task, the two dimensional relative location of the target is represented in the hidden layer by integrating the visual signal, and the representation changes adaptively according to the motion character of the robot. In the "going to a target with an obstacle" task, it is confirmed that the state that the target object hides behind the obstacle not depending on the target object, is represented in the hidden layer[5]. That is somewhat abstract information emerged by necessity of the task accomplishment. Once the hidden representation is obtained, the learning is done on this global space, and that makes the learning drastically faster.

3. ACTOR-CRITIC ARCHITECTURE

Here, actor-critic architecture[9] is employed, and actor (action command generator) and critic (state evaluator) are composed of one layered neural network. This means that the hidden layer is used by both actor and critic. This architecture is the same as the simulations in [5]. TD (Temporal Difference) is applied for the learning of the critic. TD error is defined as

$$\hat{r}_t = r_t + \gamma P_t - P_{t-1}, \tag{1}$$

where γ : a discount factor, r_t : reward, P_t : state evaluation value. The evaluation value at the previous time P_{t-1} is trained by the training signal as

$$P_{s,t-1} = P_{t-1} + \hat{r}_t = r_t + \gamma P_t, \tag{2}$$

where $P_{s,t-1}$ is the training signal for the evaluation value. On the other hand, the motion command of the robot is the sum of the outputs of $\mathbf{a}_t$ and random numbers $\mathbf{rnd}_t$ as trial and error factors. The motion command $\mathbf{a}_{t-1}$ is trained by the training signal as

$$\mathbf{a}_{s,t-1} = \mathbf{a}_{t-1} + \hat{r}_t \ \ \mathbf{rnd}_{t-1}. \tag{3}$$

The neural network is trained by Back Propagation according to Eq(2) and (3). By this learning, motion commands are trained to gain more evaluation value. Here, the layered neural network has one hidden layer and 3 output units. One of the outputs is for critic, and the other two are for actor. The output function of each hidden or output neuron is sigmoid function whose output range is from -0.5 to 0.5.

4. EXPERIMENTAL SYSTEM AND ENVIRONMENT

Fig.3 shows the robot with a monochrome visual sensor (Khepera and K213 Vision Turret) used in this paper. The specifications of Khepera and K213 Vision Turret are as follows.

Height : **55mm**
Diameter : **33mm**
Interface with PC : **RS232C(serial port)**
Transmission rate : **38400 bps**
Sensor cell : **64**
Resolution : **256 gray scale**
Visual field : **36 degree**

This visual sensor is composed of two parts (Fig.4), image perception optics and light intensity detector optics. The light optics detects light intensity around the robot at first, and then image perception optics adjusts image sensory outputs according to the light intensity. Therefore, when the light intensity is not strong enough, all the pixel values become almost white, and as a result, the robot could not distinguish bright points and dark points. In the other words, when the target is located just in front of and on the right side of the robot, the robot loses the target.

Fig.5 shows experimental environment. The action area has 70×70cm which is surrounded by a height of 10cm white paper wall, and a fluorescent light is set to keep enough light intensity. The target in the task stands 8cm tall with a diameter of 2.5cm which is wrapped black paper around.

5. APPLICATION TO A REAL ROBOT

5.1. Coping with a time delay

When Direct-Vision-Based RL is applied to a real robot, a time delay should be considered, while it does not have to be considered in simulations. PC receives visual sensory signals from the real robot through RS232C serial port, and

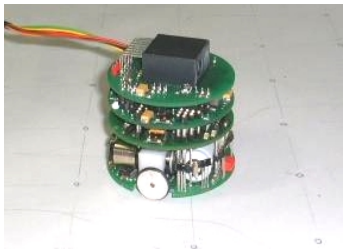

Figure 3: A picture of Khepera with K213 Vision Turret.

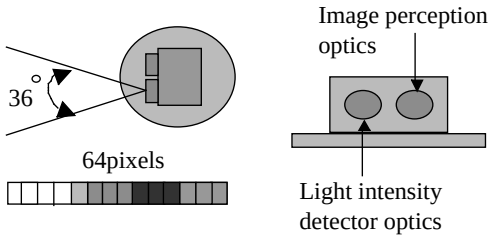

Figure 4: K213 Vision Turret.

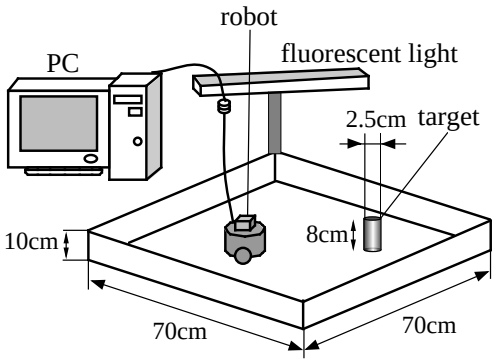

Figure 5: Experimental environment.

its transmission rate is not fast enough. The necessary time to execute each command is as follows.

Transmission of visual sensory signals : **90msec**
Transmission of action commands: **10msec**
Computation of neural network : **less than 1msec**
(for both forward and learning)

Considering the measurement interval of the visual sensor, sampling time is set to be 300msec. If computation of the neural network and transmission of the action command are done just after the transmission of the visual sensory signals, the robot continues to move with the previous action commands during the transmission of the visual sensory signals. Then, the robot location obtained from the visual sensory signals is different from the robot location when the next action command is transmitted. Here, in order to reduce this influence, the visual sensory signals are transmitted just

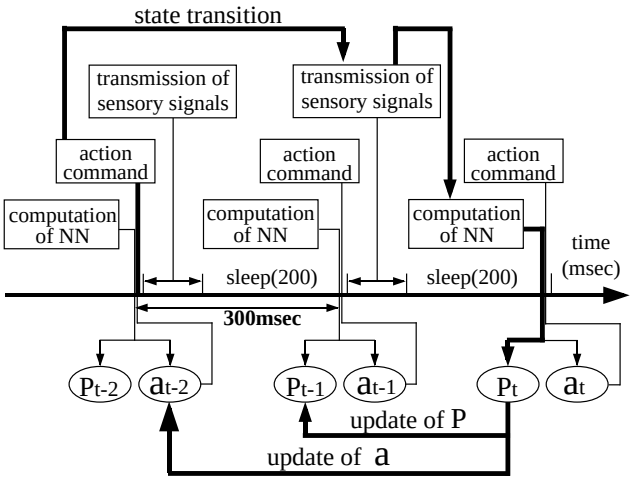

Figure 6: Timing chart of the learning for the real robot.

after the action command. Fig.6 shows the timing chart of updating of each value and system events in this experiment. Therefore, P_t is influenced by the action $\mathbf{a}_{t-2}$ on behalf of $\mathbf{a}_{t-1}$. The learning of critic is done by Eq(2) that is the same as the simulation. On the other hand, the action command at two steps before is trained by the training signal as

$$\mathbf{a}_{s,t-2} = \mathbf{a}_{t-2} + \hat{r}_t \ \ \mathbf{rnd}_{t-2} \tag{4}$$

on behalf of Eq(3).

5.2. Discrete actions

Since the action command for each wheel of Khepera should be an integer, the continuous action value is divided into an integer as

$$speed_t = (int) \ \ 4 \cdot (2 \cdot \mathbf{a}_t + \mathbf{rnd}_t), \tag{5}$$

$$if(speed_t \leq -3) \ \ speed_t = -3$$
$$if(speed_t \geq 3) \ \ speed_t = 3$$

$-3 \leq speed_t \leq 3$, $-0.5 \leq output_t \leq 0.5$,
$-0.4 \leq \mathbf{rnd}_t \leq 0.4$, where $speed$: action command for the robot.

6. EXPERIMENT

6.1. Task

In this paper, the task that the real mobile robot with monochrome visual sensor reaches a target, is employed. Here, 3-layered neural network has 64 input units, 30 hidden units, and 3 output units. One of the outputs is for critic, and the other two are for actor. Before learning, the input-hidden connection weights are small random numbers, and all the

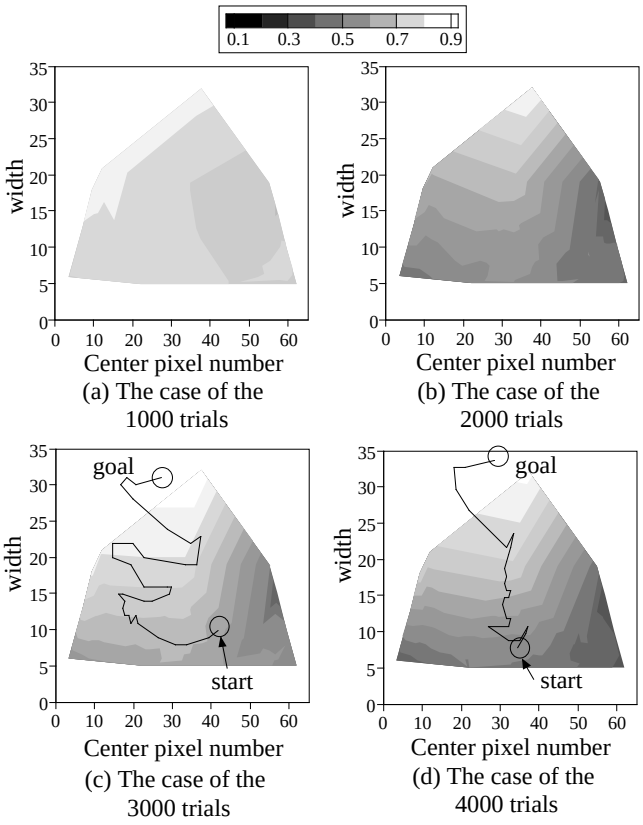

Figure 7: Distribution of evaluation value and the trajectory of the robot. These figures are drawn on the coordinates of $center\ pixel$ and $width$.

hidden-output connection weights are 0.0. After the transmission of the visual sensory signals, each of them is binalized with the boundary value of 85, and the number of pixels of the dark area is defined as $width$ of the target in the robot's view. The central pixel number of the dark area is defined as $center\ pixel$. In learning, initial $width$ and $central\ pixel$ is chosen randomly from $5 \leq width \leq 29$ and $5 \leq center\ pixel \leq 59$. From the initial position, the robot can always get the whole object on its visual sensor. Then, the robot can go to the initial position by itself according to a given program. At the beginning of learning, since the robot moves only according to the random numbers, the robot is located within the range that is close to the target. As the learning progresses, the range of the initial robot location becomes wider gradually. When $30 \leq width$ and $21 \leq center\ pixel \leq 41$, the state evaluation output is trained to be 0.4 as a reward. When the target disappears out of the visual field, it is trained to be -0.4 as a penalty. Otherwise each trial is stopped at 150 time step even if the robot can not reach the target object. The evaluation value is the sum of the evaluation output and 0.5, and the discount factor is 0.99.

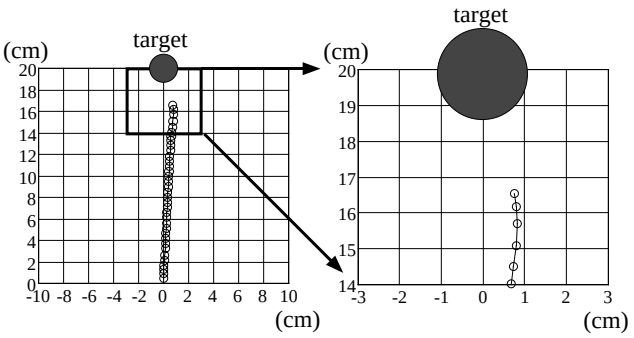

Figure 8: The locus of the robot on the field after 4000 trials.

6.2. Learning result

Fig.7 shows the state evaluation value after learning. This figure is drawn by computing the outputs off-line for 35 sample sets of visual sensory signals. The vertical axis indicates the $width$ and the horizontal axis indicates the $center\ pixel$ of the target object on the robot's view. It is seen that as the learning progress, the distribution of the state evaluation value is formed gradually. As shown in Fig.7, the state evaluation value becomes larger when the $width$ of the target becomes larger on the visual sensor. It is smaller when the target is shown on the right side of the visual sensor than on the left side. The reason is that the robot misses the target

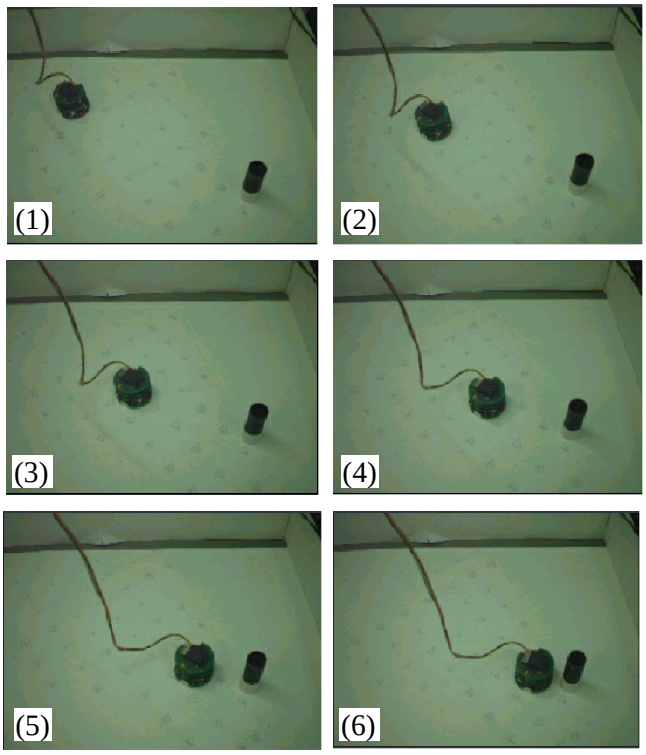

Figure 9: The robot succeeded in reaching a target object.

when it is shown on the right side of the sensor as mentioned in the section 4. Fig.7 (c), (d) show the locus of the target object in the robot's view, and the robot approaches the target while looking at it on the left side of the visual sensor.

Next, Fig.8 shows the locus of the robot on the absolute coordinates after 4000 trials (the same trial as Fig.7 (d)). When the robot comes near the target object, the robot catches the target object on the center of the robot's view by rotating counterclockwise. Fig.9 shows a time series of photos to show that the robot reaches the target object without missing. As the result of the learning, the robot could obtain the action to overcome the defect of the visual sensor.

7. CONCLUSION

Direct-Vision-Based RL was applied to a real robot with a linear and monochrome visual sensor. Considering the time delay to get the visual sensory signals, it was proposed that the actor outpus are trained using the critic output at two time steps ahead. It was shown that the robot with a monochrome visual sensor could obtain reaching actions to a target object through the learning from scratch without any advance knowledge and any helps of humans. The robot could obtain the actions to avoid missing of the target that happens due to a sensor character.

8. REFERENCES

[1] Anderson, C. W. (1989) "Learning to Control an Inverted Pendulum Using Neural Networks", *IEEE Control System Magazine*, Vol. 9, pp.31-37

[2] Morimoto, J. and Doya, K. (2001) "Acquisition of Stand-up Behavior by a Real Robot using Hierarchical Reinforcement learning ", *Robotics and Autonomous Systems*, Vol. 36, pp.37-51

[3] Tesauro, G. J. (1992) "Practical Issues in temporal difference learning", *Machine Learning*, 8, pp.257-277

[4] Asada, M., Hosoda, K. and Suzuki, S. (1996) "Vision-based Behavior Learning and Development for Emergence of Robot Intelligence",*The 7th Int'l Sympo. on Robotics Research*, pp.327-338

[5] Shibata, K., Ito, K. and Okabe, Y.(1998) "Direct-Vision-Based Reinforcement learning "Going to a Target" Task with an Obstacle and with a Variety of Target Sizes", *Proc. of NEURAP'98*, pp.95-102

[6] Shibata, K., Sugisaka, M. and Ito, K. (2001) "Fast and Stable Learning in Direct-Vision-Based Reinforcement learning", *Proc. of the 6th AROB (Int'l Sympo. on Artificial Life and Robotics)*, Vol.1, pp.200-203

[7] Maehara, S., Sugisaka, M. and Shibata, K. (2001) "Reinforcement Learning Using Gauss-Sigmoid Neural Network", *Proc. of AROB 6th*, Vol. 2, pp.562-565

[8] K. Shibata and K. Ito (1998) "Reconstruction of Visual Sensory Space on the Hidden Layer in a Layered Neural Networks", *Proc. of ICONIP (Int'l Conf. on Neural Information Processing) '98*, Vol. 1, pp.405-408

[9] Barto, A. G., Sutton, R. S. and Anderson, C. W. (1983) "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems", *IEEE Trans. SMC-13*, pp.835-846