Acquisition of active perception and recognition through Actor-Q learning using a movable camera

Ahmad Afif Mohd Faudzi* † and Katsunari Shibata*,
* Department of Electrical and Electronic Engineering, Oita University
† Department of Electrical and Electronic Engineering, Universiti Malaysia Pahang, Malaysia
Email: afif9903@gmail.com shibata@cc.oita-u.ac.jp

***Abstract*—In the previous work, it was shown in simulation that by employing Actor-Q learning, active perception and recognition problem can be learned. In this learning, a camera with non-uniform resolution was assumed. Appropriate camera motions and recognition timing as well as the correct recognition were acquired through the reinforcement learning using the final recognition results as rewards and punishments[1]. In this paper, the authors tried to verify using a real camera whether appropriate continuous camera motions and identification of two patterns can be acquired simultaneously through Actor-Q learning. Here, non-uniform resolution was also applied. As a result, within a limited number of trials, these functions cannot be achieved. Then, three stages of learning are introduced. The system was trained on "camera motion" task at first and then trained on "pattern recognition" task. After that the system was trained on both "pattern recognition" and "camera motion" simultaneously. It was verified that appropriate camera motion, recognition and recognition timing can be learned only from the reward and punishment through Actor-Q learning.**

***Keywords*—reinforcement learning, neural network, active perception, pattern recognition, Actor-Q learning.**

I. Introduction

Acquiring external information is one of the most important functions for biological subjects including humans, and the information can be obtained through various sensory organs. However, in the real world, there is a huge amount of external information available, which makes it difficult and less efficient when acquiring all the information in details. In order to cope with this situation, biological subject actively moves the sensory organ and efficiently acquires only necessary information. This function is called active perception.

Vision is probably the most developed perception in humans and acquires the largest amount of information among the human sensory organs. The allocation of the sensor cells on our eye retina is non-uniform. It seems that, to make correct recognition efficiently, we use the whole sensor to acquire the external information roughly, and we will move the dense part to the direction of the object that we think valuable.

The eye movement and recognition in humans seem very flexible and intelligent. Let us consider the case when we read a book. We do not seem to read it by recognizing each character one by one. We would predict a word from the first two or three characters, and would utilize the story to expect the next word or character. It does not seem that there is a simple formulaic routine about the way of moving the eyes, extracting individual character, or recognizing patterns or objects. We consider many things simultaneously, and move our eyes and recognize the story very flexibly. Such parallel consideration and flexibility must be achieved by our parallel and flexible brain, and learning must play an important role for it.

The investigation on automatic visual tracking systems has been done for a long time, and now some of them, e.g., security cameras are being used widely. Recent robots, such as ASIMO and AIBO also have a camera(s). However, an algorithm to move the camera using pattern matching, optical flow or some other technique is provided by humans. As far as the authors know, there is no system that acquires not only recognition function, but also the camera motion function flexibly through learning.

Research on character or object recognition also has been going on for a long time[4]. Nowadays, hand-written character recognition systems have been practically used already in post offices and so on. The recognition rate is high. However, these systems were usually limited to the cases when only one character is written in a predetermined area. Therefore, character segmentation still remains as one of the big problems in image recognition[5]. When a pattern on an image is shifted, the image signals are completely different from the previous signals. Therefore, even though small shift or size change can be absorbed[6], it is inefficient to develop a recognition system that can recognize the target patterns for any views. It is more efficient to move the camera to the position where it can recognize the target pattern easily.

Recently, the coupling of a neural network (NN) and reinforcement learning (RL) is considered useful in machine learning because of its autonomous, adaptive and flexible learning ability. However, many researchers are still positioning RL as a learning specific for actions in the total process, and the NN as a non-linear function approximator. On the other hand, there is an approach, where by applying RL to a neural network that bears the entire process from sensors to motors, the necessary functions including recognition and memory emerge[2][3]. Flexible sensor motion also can be expected to emerge through this framework of learning.

Under this concept, active perception and recognition learning system based on Actor-Q learning has been proposed[1]. A visual sensor with non-uniform sensor cell was used. This learning was done by simulations and it was successfully verified that camera motion, recognition, and recognition tim-

ing can be learned simultaneously and flexibly only from the recognition results as rewards and punishments.

In this paper, the authors would like to verify using a real camera that the active perception and recognition function can also be achieved by the learning with Actor-Q method. It is examined whether flexible camera motion and recognitions including recognition timing can be obtained through learning. Visual image with non-uniform resolution is used as NN input signals and the camera motion is controlled by the output signals. However, because a regular layered NN is used here, the context-based intelligent recognition that mentioned above cannot be expected here, and remains as a future work.

The remainder of this paper is organized as follows. Section 2 describes the learning system. Section 3 presents the task problem and system settings. Experimental results and analysis are presented in Section 4. Finally, we conclude and describe future works in Section 5.

II. Learning System

Actor-Q learning[1] enables a learner to learn simultaneously both discrete decision making and continuous motions generation. In this learning system, the outputs are divided into two types; "actions", which are discrete intentions, and "motion", which is a vector with continuous values. For every step, "action" is determined, and if the action required "motion", "motion" is also determined. "Action" is chosen based on Q-values and Q-learning is used for training. On the other hand, actor, which is usually used in Actor-Critic learning, is employed to generate a "motion". In this system, the Q-value corresponding to the "motion" is used to train the actor on behalf of the critic.

As shown in Fig. 1, this learning system has two neural networks (NNs) called Q-net and Actor-net respectively. The Q-net is responsible to choose an "Action" and the Actor-net is responsible for controlling camera motion. In this system, "to output the recognition result" and "to move the camera" are considered as actions. When there are P patterns to be presented, there are $(P+1)$ possible actions, which include P actions to give the recognition result for respective patterns, as well as an action to move the camera. Each output of Q-net is used as a Q-value after linear transformation.

If "recognition" is selected as an action, it is examined whether the result is correct or not. A reward is given and it completes the trial. The training signal for "recognition" action "$Recog:p$" is given as

$$Q_{\text{“}Recog:p\text{”},train} = r \tag{1}$$

where r is a reward, which is set to 0.9 for a correct result, and 0.1 for an incorrect result.

When the action "camera motion" is selected, the camera moves according to the Actor-net output. In this case, the trial is not completed. New input signals obtained after the camera movement, and the next action is selected. The training signal for "camera motion" action is given as

$$Q_{\text{“}motion\text{”},train} = \gamma max_{a \in A} Q_a(\mathbf{s}_{t+1}) \tag{2}$$

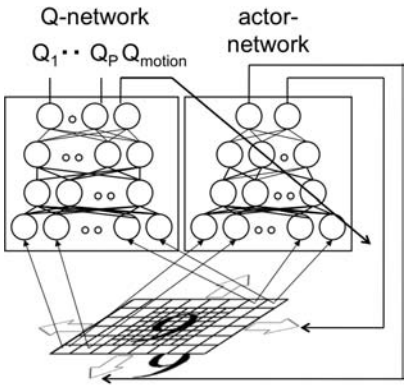


Fig. 1. Actor-Q based active perception and recognition learning system.

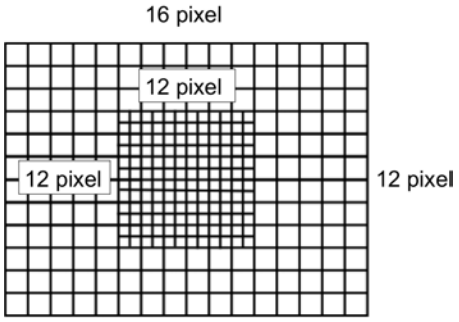


Fig. 2. Visual image with non-uniform resolution.

where γ is a discount factor and A is the possible action set.

The camera movement $\mathbf{m}$ specifies the camera angle displacement in x- and y- directions. It is given as

$$\mathbf{m} = K(\mathbf{O}_m + \mathbf{rnd}) \tag{3}$$

by adding a random vector $\mathbf{rnd}$ to Actor-net output $\mathbf{O}_m$ and multiplied by a constant K. The training signal for Actor-net output $\mathbf{O}_m$ is given as

$$\mathbf{O}_{m,train} = \mathbf{O}_m + (\gamma max_a Q_a(\mathbf{s}_{t+1}) - Q_{\text{“}motion\text{”}}(\mathbf{s}_t))\mathbf{rnd}. \tag{4}$$

The training for both Q-net and Actor-net is executed by Back Propagation (BP) using the above training signals.

III. Experiment

A. Task Problem

In this research, a PTZ(Pan Tilt Zoom) camera (Canon VC-C50i) as a visual sensor and a monitor to display patterns were used. Two patterns are prepared, and the system needs to identify which of the patterns is presented. As shown in Fig. 2, non-uniform resolution is introduced, where the central part of the sensor has a high-resolution and the peripheral part has a low resolution.

In the learning process, the presented pattern was chosen randomly for every trial. The initial position of the camera is also different in each trial, and no information about the initial position of the camera is given to the NN. As shown

(a) (b)

Fig. 3. The difference of the captured image depending on the part that captures a presented pattern.

in Fig. 3(a), when a pattern is captured at the peripheral part, it is hard to make the identification even the whole pattern is in the visual field. As a result, the system needs to move the camera to an appropriate position. In this case, the camera has to move to bottom-left direction until the pattern is captured around the center of the image as in Fig. 3(b), and finally to decide which of the patterns is presented. The reinforcement signal is given according to whether the final recognition result is correct or not. Thus, the camera motion, recognition and the timing to output the recognition result are acquired only through reinforcement learning.

B. System Setting

There are two patterns used in this experiment, pattern '0' and pattern '9'. The original captured uniform image is 640×480 pixels. However, as shown in Fig. 2, it is converted to non-uniform resolution with 300 pixels. As shown in Fig. 1, the learning system has two NNs and every NN has 4 layers. The inputs are the raw image signals (300 pixels $\times$ 3 (RGB)=900) that are linearly converted to a value between -0.5 and 0.5. The levels of difficulty are introduced to accelerate learning. The initial position of the camera is set randomly for each trial within the range that depends on the level. Since the ranges were set differently depends on the task, it is described later.

For the Q-net, there are three output neurons. Each of them represents the Q-value corresponding to one of the three actions, which are two "recognition results" and "camera motion". The number of neurons in each layer is 900-250-20-3 from the input layer to the output layer. On the other hand, there are two outputs neurons for the Actor-net that controls the camera movement called pan (x-axis) and tilt (y-axis). The number of neurons in each layer is 900-250-20-2 from input to output layer.

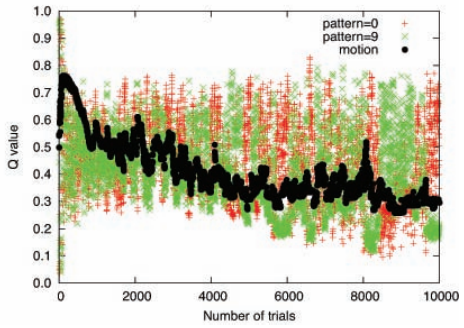
The output function of each neuron is the sigmoid function with the range from -0.5 to 0.5, that is, (1/(1+exp(-x))-0.5). In the Q-net, all the outputs are used after adding 0.5. When the correct result is outputted, the corresponding Q-value is trained to be 0.9, and 0.1 when it is not correct. This corresponds to 0.4 and -0.4 for the training signals of the NN respectively. If the training signal for each output neurons is less than -0.4 or more than 0.4, the training signal is set to be -0.4 or 0.4 respectively to keep it remaining between -0.4 and 0.4 and to avoid the saturation range of the sigmoid function.

The initial values of connection weight from hidden to output are all 0.0. Those from input to hidden are set to a random number between -0.1 and 0.1. The discount factor γ in the learning of Q-value by Eq.(2) and Eq.(4) is set as 0.986. In Eq.(3), the maximum movement by the camera **m** in a unit time is 2.25° in both x- and y- directions. The random variable **rnd** is derived by an uniform random number powered by 3.0. The value range is -1.0 to 1.0 at first, and is gradually reduced according to the progress of learning.

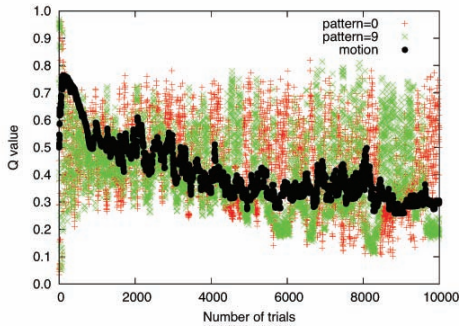
In action selection, ε-greedy selection is used during learning, where the value of ε is gradually decreased as

$$\varepsilon = exp(-trial \times 0.00017) \tag{5}$$

where $trial$ is the number of current trial. ε is compared with a random number between 0.0 to 1.0. If it is larger than the random number, the action with the maximum Q-value is selected greedily.



(a) Pattern 0



(b) Pattern 9

Fig. 4. The change of Q-values at the end of every trial in the learning of active perception and recognition task from scratch.

IV. Learning Results

As a result, within 10,000 trials, active perception and recognition function cannot be achieved. Fig. 4 shows the Q-value for each action at the end of every trial (a) when the pattern '0' was presented and (b) when the pattern '9' was presented respectively.

If the learning proceeds successfully, for example, in graph (a), the red point (+), which represents the Q-value for the recognition output '0', should increase gradually and separate from the green point ($\times$), which represents the Q-value for the recognition output '9'. However, comparing with (a) and (b), the change of Q-value is almost the same, not depending on the presented pattern, even though the recognition result changes alternately during learning. The black point ($\bullet$), which

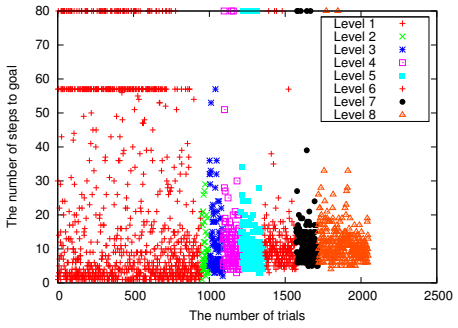


Fig. 5. Learning curve for the "camera motion" task.

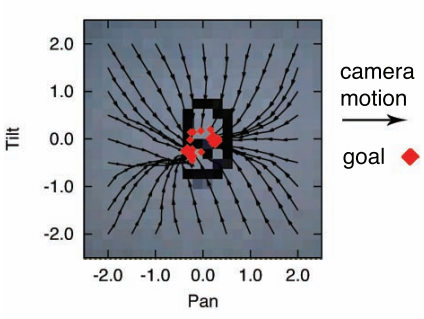
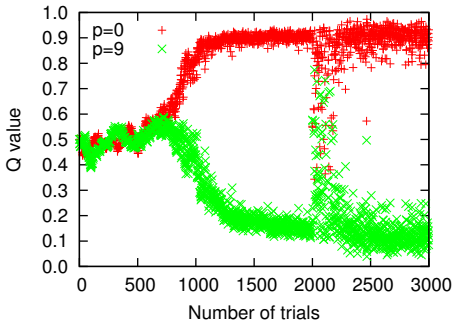
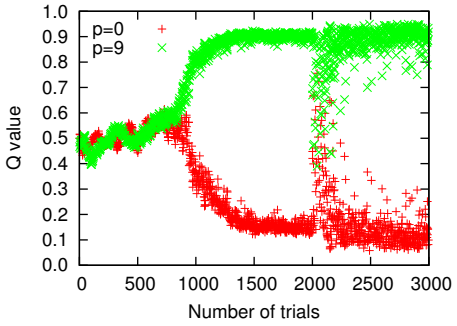


Fig. 6. Some sample camera trajectories after learning of "camera motion" task.



(a) Pattern 0



(b) Pattern 9

Fig. 7. The change of Q-values in the learning of the "recognition" task.

corresponds to the "camera motion", gradually decreased. This might happen because the system could not find the area where the presented pattern can be identified correctly.

In the previous research [1], the number of trials reached as large as 100,000. Comparing to this number, more number may enable the learning to be successful. However, since the real camera is used, it is difficult to do a large number of trials while considering the load of camera motors and the time spent for learning.

Then, three stages of learning including two preliminary learning for "camera motion" and "pattern recognition" are introduced. First, the system was trained on "camera motion" task. In this task, one of the two patterns was presented on a monitor as a target object. When the system moves the camera to the place where the target object is caught in the range of $\pm 1.7^\circ$ from the center of the image, a reward is given. For every trial, the number of steps was limited to 57 steps and if the target object disappeared out of the visual field, the trial terminated with setting the number of steps to 80. The initial position for every trial was set randomly with a certain range, which becomes wider gradually depending on the level. In order to go to the next level, the system must succeed in successive 50 trials. The range in the level 1 is $\pm 4^\circ$ for each of pan and tilt. In the level 8 that is the final level, the range is $\pm 11^\circ$, but the area of $\pm 10^\circ$ is excluded. Here, only Actor-net and Q-value corresponding to "camera motion" were trained just like the learning of the regular actor-critic.

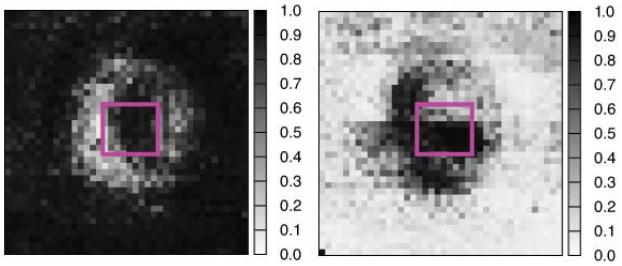
Fig. 5 shows the learning curve of this task, which plots the number of steps to achieve the task for every trial. The learning process finished successfully at around the 2000th trial. For the level 1(red points), it takes about 1000 trials until the camera can catch the target around the center within the limited number of steps. After that, when the level increases, the number of steps increases temporary, but the system accomplishes the level in a smaller number of trials. As the level increases, the optimal number of steps to succeed in the task also increases, and it is around 6 to 15 steps for the level 8. After that, the random vector for camera motion was removed, and the system was tested. Fig. 6 shows the camera trajectory from 32 initial positions that are in the range of the level 8. It is seen that the system is capable to move the camera appropriately until the target object caught around the center of the image.

The "camera motion" learning was followed by "pattern recognition" learning. Patterns were presented at the center of the image. After 2000 trials, the camera position was set randomly with a very small range as $\pm 2.5^\circ$ from the home position. Here, only Q-values corresponding to "pattern recognition" in Q-net were trained. Fig. 7 shows the change of Q-values when (a) pattern '0' and (b) pattern '9' were presented. It shows that, when (a) pattern '0' was presented, the Q-values correspond to pattern '0' (red points) and pattern '9' (green points) is started to separate around 800 trials. When the position of the camera was shifted after the 2000th trial, the Q-value is not stable at first. However, after 500 trials, the system can recognize it correctly even though some variation remains. It goes the same with Fig. 7(b).

Fig. 8 shows the distribution of the Q-value for the presented

TABLE I
LEVEL OF DIFFICULTY

Number of trials	Level	Range of angle from home position
0 to 5500	1	$\pm 2.5^\circ$
5501 to 8000	2	$\pm 3.0^\circ$
8001 to 10000	3	$\pm 3.5^\circ$



(a) Pattern 0 (b) Pattern 9

Fig. 8. Distribution of Q-value for each presented pattern according to the camera postion.

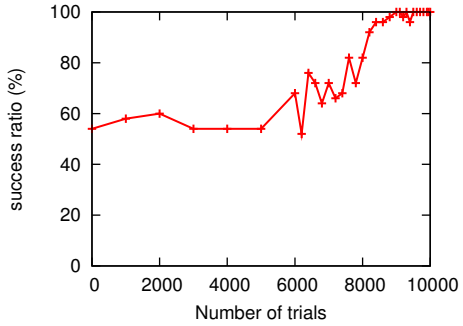
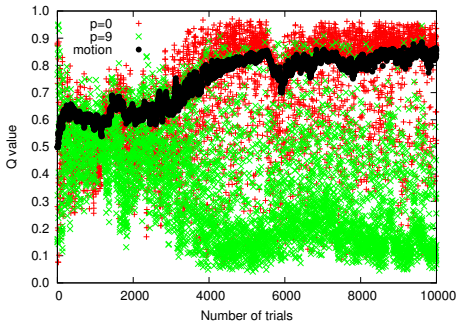


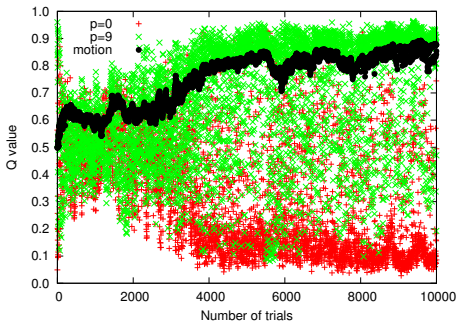
Fig. 9. Learning curve for active perception and recognition task after two preliminary learnings.

pattern according to the camera position for each presented pattern. It is seen in this result that the corresponding Q-value is large in both patterns when the camera is in the range where the system learned in the learning phase. In other range, if the corresponding Q-value is large in one pattern, it is not large in the other pattern.

Finally, in the 3rd stage, the whole Actor-Q system was trained on both "camera motion" and "pattern recognition" in parallel. Here, the initial position of the camera is also set randomly for each trial within the range that depends on the stages level. As shown in Table 1, there are three levels of difficulty. The level increases when the number of trials reaches to certain counts, and is applied to both pan and tilt. During learning, the connection weight set was stored at every 1,000 trials. Fig. 9 shows the learning curve of this task. The vertical axis indicates the ratio of the correct recognition over 50 trials using the stored connection weight set with no random factors. At the beginning, the success ratio is as low as around 50%. At this time, the system could not decide the recognition result correctly because, in the first two stages of learning,



(a) Pattern 0



(b) Pattern 9

Fig. 10. The change of Q-values at the end of every trial in the learning of active perception and recognition task after two preliminary learnings.

the Q-values for recognition result was not trained for the case when the pattern is not caught around the center of the image. However, the success ratio increases rapidly between 6000 and 8000 number of trials. Finally, around 10000 trials, the probability of correct recognition reaches 100%. Note that the success ratio was not so high when the test is performed in daytime. That would be because the learning was performed only at night, and the system learned only for night time.

Fig. 10 shows the Q-values for the actions at the end of every trial for each presented pattern during learning in the case of the parallel learning. When the pattern '0' was presented, Fig. 10(a) shows that the red point (+), which is the Q-value corresponding to the pattern '0', increased and became higher than the other Q-values. At the same time, the green point ($\times$) that corresponds to the pattern '9' was decreased. By these big differences of Q-values, the system can make recognition correctly. However, there are some green points that mixed with the red points until the learning finished. It happened because the effect of the ε-greedy that was used in the selection of actions. At 10000th trials, the probability of action selection chosen by the random factor is 18%.

Fig. 11 shows a series of images from an initial position to the final position where the system decided to output the recognition result. It is shown that, in the step 4, the presented pattern was completely captured in the high-resolution part of the image. After that, the camera only makes some fine movement for searching an appropriate position and finally outputted the recognition result. Fig. 12 shows the change of

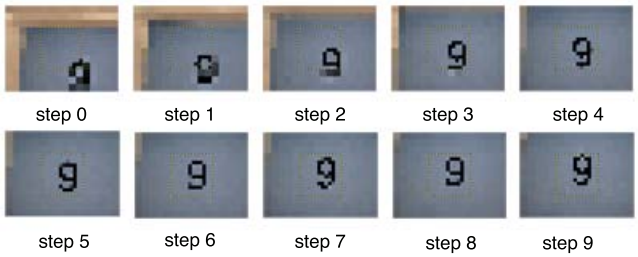


Fig. 11. Learning curve for active perception and recognition task after two preliminary learnings.

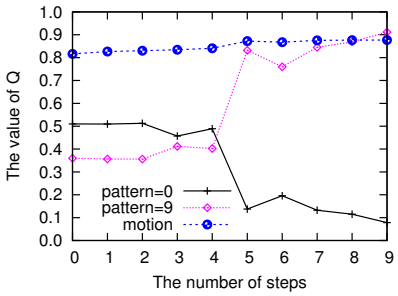


Fig. 12. The change of Q-values in the trial as shown in Fig. 11.

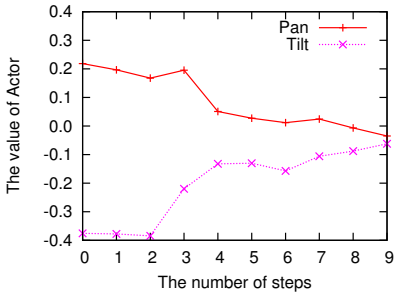


Fig. 13. The change of actor outputs in the trial as shown in Fig. 11.

Q-values in the case of Fig. 11. From the beginning, the Q-value for "camera motion" (•) is clearly high, but, Q-value for "recog: p=9" (◇) is getting closer. Finally, the Q-value for "p=9" becomes larger than the Q-value for "camera motion", and the recognition result "p=9" was outputted. At the same time, the Q-value for "p=0" (+) starts to separate from the Q-value for "p=9" at the step 5. It might be expected that additional learning enables the system to respond the correct recognition result earlier. In Fig. 13, we can see the actor outputs change in order to bring the target object to the center. The positive value of Pan and the negative value of Tilt indicate that the system moves the camera to the bottom-right direction as shown in Fig. 11.

Fig. 14 shows the camera trajectories from 24 initial camera positions. It is seen that, even though the presented pattern and initial position were different, the system moves the center of the camera almost to the same positions (◇). Compared to the Fig. 6 in the case of "camera motion" learning, this movement almost the same, but the final destination is different. Fig. 15

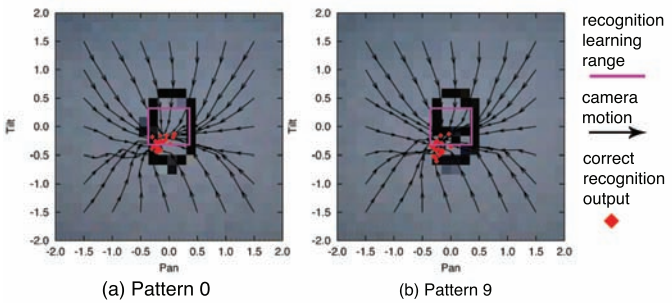


Fig. 14. The camera trajectory and the point where the system decided to output the correct recognition result for each pattern.

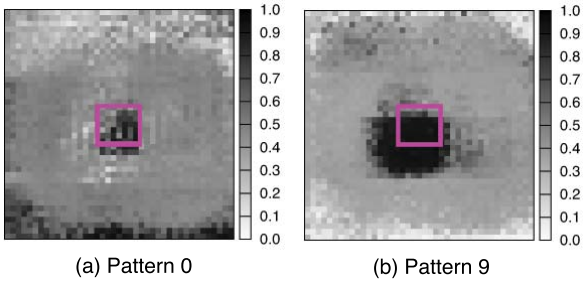


Fig. 15. Distribution of Q-value for the presented pattern according to the camera position after learning.

shows the distribution of Q-value for each presented pattern. Comparing to Fig. 8, it is seen that there are big changes of Q-value distribution especially outside of learning range in case of "pattern recognition" learning. The corresponding Q-value is obviously large at bottom-left part of the center in case of the pattern '9' compared to pattern '0'.

From all these results, it is thought that these two preliminary learning; "camera motion" and "pattern recognition" play an important role in this learning. The third learning harmonized "camera motion" and "pattern recognition", and realized efficient active perception and recognition.

V. Conclusion

In this paper, active perception and recognition learning based on Actor-Q method was verified using a real movable camera. A non-uniform resolution was applied. However, within a limited number of trials, this function cannot be achieved. Then, as an alternative way, three stages of learning are introduced. The authors trained the system on "camera motion" task and followed by trained on "pattern recognition" tasks separately. Then, the whole system was trained on both "camera motion" and "pattern recognition" simultaneously. The learning was successful. The system moves the camera to the appropriate location and output the correct recognition result. The appropriate camera motion, recognition, and recognition timing were successfully acquired through learning.

We are currently working on using of more patterns, in order to investigate the system performance in more difficult task. In addition, to increase the performance in daytime, the learning needs to be done on both daytime and night. Furthermore, we

are considering to apply the recurrent neural network in order to acquire context-based intelligent motion and recognition.

ACKNOWLEDGMENT

This research was supported by JSPS Grant in-Aid for Scientific Research #19300070. Special thanks to all members of Shibata Laboratory, Oita University for all supports.

REFERENCES

[1] K. Shibata, T. Nishino, and Y. Okabe, *Active Perception and Recognition Learning System Based on Actor-Q Architecture*, System and Computers in Japan, Vol. 33, No. 14, pp. 12-22, 2002.

[2] K. Shibata and T. Kawano *Acquisition of Flexible Image Recognition by Coupling of Reinforcement Learning and a Neural Network*, SICE Journal of Control, Measurement, and System Integration (JCMSI), Vol. 2, No. 2, pp. 122-129, 2009.

[3] H. Utsunomiya and K. Shibata *Contextual Behavior and Internal Representations Acquired by Reinforcement Learning with a Recurrent Neural Network in a Continuous State and Action Space Task*, Advances in Neuro-Information Processing, Lecture Notes in Computer Science, Proc. of ICONIP (Int'l Conf. on Neural Information Processing) 08, Vol. 5507, pp. 970-978, 5507-0970.pdf (CD-ROM), 2009.

[4] A. Iwata *Hand-written alphanumeric recognition by a self-growing neural network "CombNET-2"*, Proc. of Int. Joint Conf. on Neural Network,Vol.4, pp.228-234, 1992

[5] Mei-Sen Pan, Jun-Biao Yan and Zheng-Hong Xiao *Vehicle license plate character segmentation*, International Journal of Automation and Computing, Vol. 5, No. 4, pp. 425-432, 2008.

[6] K. Fukushima *Neocognitron : A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*, Boil. Cybenetics 36(4), pp. 193-202, 1980.