Spiral Development of Behavior Acquisition and Recognition Based on State Value

Yasutake Takahashi[1], Yoshihiro Tamura[1], and Minoru Asada[1,2]

[1] Dept. of Adaptive Machine Systems, Graduate School of Engineering
Osaka University
[2] JST ERATO Asada Synergistic Intelligence Project
Yamadaoka 2-1, Suita, Osaka, 565-0871, Japan
{yasutake,yoshihiro.tamura,asada}@ams.eng.osaka-u.ac.jp
http://www.er.ams.eng.osaka-u.ac.jp

Abstract. Both self-learning architecture (embedded structure) and explicit/implicit teaching from other agents (environmental design issue) are necessary not only for one behavior learning but more seriously for life-time behavior learning. This paper presents a method for a robot to understand unfamiliar behavior shown by surrounding players through the collaboration between behavior acquisition and recognition of observed behavior, where the state value has an important role not simply for behavior acquisition (reinforcement learning) but also for behavior recognition (observation). That is, the state value updates can be accelerated by observation without real trials and errors while the learned values enrich the recognition system since it is based on estimation of the state value of the observed behavior. The validity of the proposed method is shown by applying it to a soccer robot domain.

1 Introduction

Reinforcement learning has been studied well for motor skill learning and robot behavior acquisition in both single and multi-agent environments. Especially, in the multi-agent environment, observation of surrounding players make the behavior learning rapid and therefore much more efficient [1,3,7]. Actually, it is desirable to acquire various unfamiliar behavior with some instructions from others in real environment because of huge exploration space and enormous learning time to learn. Therefore, behavior learning through observation has been more important. Understanding observed behavior does not mean simply following the trajectory of an end-effector or joints of demonstrator. It means reading his/her intention, that is, the objective of the observed behavior and finding a way how to achieve the goal by oneself regardless of the difference of the trajectory. From a viewpoint of the reinforcement learning framework, this means reading rewards of the observed behavior and estimating sequence of the value through the observation.

Takahashi et al.[6] proposed a method of not only to learn and execute a variety of behavior but also to recognize observed behavior executed by surrounding

L. Iocchi et al. (Eds.): RoboCup 2008, LNAI 5399, pp. 343–354, 2009.

players supposing that the observer has already acquired the values of all kinds of behavior the observed agent can do. The recognition means, in this paper, that the robot categorizes the observed behavior to a set of its own behavior acquired beforehand. The method seamlessly combines behavior acquisition and recognition based on "state value" in reinforcement learning scheme. Reinforcement learning generates not only an appropriate behavior (a map from states to actions) to accomplish a given task but also a utility of the behavior, an estimated sum of discounted rewards that will be received in future while the robot is taking an appropriate policy. This estimated discounted sum of reward is called "state value." This value roughly indicates closeness to the goal state of the given task if the robot receives a positive reward when it reaches the goal and zero else, that is, if the agent is getting closer to the goal, the value becomes higher. This suggests that the observer may recognize which goal the observed agent likes to achieve if the value of the corresponding task is going higher.

This paper proposes a novel method that enhances behavior acquisition and recognition based on interaction between learning and observation of behavior. Main issues to be attacked here is

– categorization of unfamiliar behavior based on reading rewards, and
– enhancement of the behavior recognition and acquisition.

A learning/recognizing robot assumes that all robots and even the human player share reward models of the behavior. For example, all robots and the human player receive a positive reward when the ball is kicked into the goal. This assumption is very natural as we assume that we share "value" with colleagues, friends, or our family in our daily life. The reading rewards during the observation of unfamiliar behavior gives a hint to categorize the observed behavior into the one that is meaningful to the observer. For the second issue, a robot learns its behavior through not only trials and errors but also reading rewards of the observed behavior of others (including robots and humans). Fig. 1 shows a rough idea of our proposed method. $V(s)$ and $\hat{V}(s)$ are the state value updated by oneself and the state value estimated through observation, respectively. Takahashi et al.[6] showed the capability of the proposed method mainly in case that the observer has already acquired a number of behavior to be recognized beforehand. Their case study showed how this system recognizes observed behavior based on the

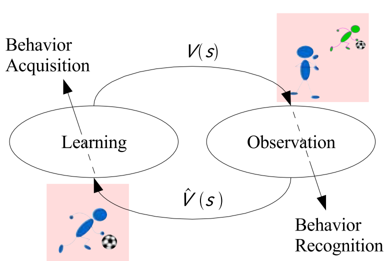


Fig. 1. Interaction between Learning and Observation of Behavior

state value functions of self-behavior. This paper shows how the estimated state value of observed behavior, $\hat{V}(s)$, gives feedback to learning and understanding unfamiliar observed behavior and this feedback loop enhances the performance of observed behavior recognition. The validity of the proposed method is shown by applying it to a soccer robot domain including a human player.

2 Outline of the Mechanisms

The reinforcement learning scheme, the state/action value function, recognition of observed behavior based on state value function, and the modular learning system for various behavior acquisition/recognition are explained, here.

2.1 Behavior Learning Based on Reinforcement Learning

An agent can discriminate a set S of distinct world states. The world is modeled as a Markov process, making stochastic transitions based on its current state and the action taken by the agent based on a policy π. The agent receives reward r_t at each step t. State value V^π, the sum of the discounted rewards received over time under execution of a policy π, will be calculated as follows:

$$V^\pi = \sum_{t=0}^{\infty} \gamma^t r_t \ . \tag{1}$$

For simplicity, it is assumed here that the agent receives a positive reward if it reaches a specified goal and zero else. Then, the state value increases if the agent comes close to the goal by following a good policy π. The agent updates its policy through trials and errors in order to receive higher positive rewards in future. Analogously, as animals get closer to former action sequences that led to goals, they are more likely to retry it. For further details, please refer to the textbook of Sutton and Barto [4] or a survey of robot learning [2].

Here we introduce a model-based reinforcement learning method. A learning module has a forward model which represents the state transition model and a behavior learner which estimates the state-action value function based on the forward model in a reinforcement learning manner.

Each learning module has its own state transition model. This model estimates the state transition probability $\hat{\mathcal{P}}^a_{ss'}$ for the triplet of state s, action a, and next state s':

$$\hat{\mathcal{P}}^a_{ss'} = Pr\{s_{t+1} = s' | s_t = s, a_t = a\} \tag{2}$$

Each module has a reward model $\hat{\mathcal{R}}_s$, too:

$$\hat{\mathcal{R}}(s) = E\{r_t | s_t = s\} \tag{3}$$

All experiences (sequences of state-action-next state and reward) are simply stored to estimate these models. Now we have the estimated state transition

Fig. 2. Robots with a human player in a Soccer Field

probability $\hat{\mathcal{P}}^a_{ss'}$ and the expected reward $\hat{\mathcal{R}}_s$, then, an approximated state-action value function $Q(s,a)$ for a state action pair s and a is given by

$$Q(s,a) = \sum_{s'} \hat{\mathcal{P}}^a_{ss'} \left[\hat{\mathcal{R}}(s') + \gamma V(s')\right] \tag{4}$$

$$V(s) = \max_a Q(s,a) \; , \tag{5}$$

where $0 < \gamma < 1$ is a discount factor[1].

2.2 Behavior Recognition Based on Estimated Values

An observer watches a demonstrator's behavior and maps the sensory information from an observer viewpoint to a demonstrator's one with a simple mapping of state variables[2]. Fig. 2 shows two robots, a human player and color-coded objects, e.g., an orange ball, and a goal. The robot has an omni-directional camera on top. A simple color image processing is applied in order to detect the color-coded objects and players in real-time. The mobile platform is based on an omni-directional vehicle. These two robots and the human play soccer such as dribbling a ball, kicking it to a goal, passing a ball to the other, and so on. While playing with objects, they watch each other, try to understand observed behavior of the other, and emulate them. Fig. 3 shows a simple example of this transformation. It detects color-coded objects on the omni-directional image, calculates distances and directions of the objects in the world coordinate of the observer, and shifts the axes so that the position of the demonstrator comes to center of the demonstrator's coordinate. Then it roughly estimates the state information of the demonstrator. A sequence of its state value from the estimated state of the observed demonstrator is estimated. If the state value goes up during an observed behavior, it means that the behavior derived by the state value system is valid for explaining the observed behavior. Fig. 4 shows an example task of navigation in a grid world and a map of the state value of the task. There is a goal state at the top center of the world. An agent can move one of the neighboring square in the grids every step. It receives a positive reward only when

[1] The discount factor represents that distant rewards are less important.

[2] For a reason of consistency, the term "demonstrator" is used to describe any agent from which an observer can learn, even if the demonstrator does not have an intention to show its behavior to the observer in this paper.

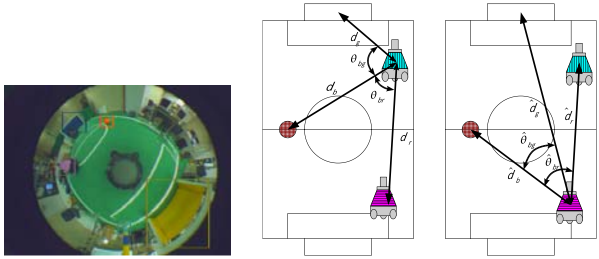


Fig. 3. Estimation of view of the demonstrator. Left : a captured image the of observer, Center : object detection and state variables for self, Right : estimation of view of the demonstrator.

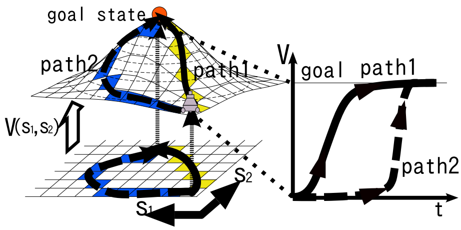


Fig. 4. Behavior recognition based on the change of state value

it stays at the goal state while zero else. There are various optimal/suboptimal policies for this task as shown in the figure. If the agent follows an appropriate policy, the value is going up even if it is not exactly the optimal one.

Here we define recognition reliability g that indicates how much the observed behavior would be reasonable to be recognized as a behavior

$$g = \begin{cases} g + \beta & \text{if } V_t - V_{t-1} > 0 \text{ and } g < 1 \\ g & \text{if } V_t - V_{t-1} = 0 \\ g - \beta & \text{if } V_t - V_{t-1} < 0 \text{ and } g > 0 \end{cases},$$

where β is an update parameter, and 0.1 in this paper. This equation indicates that the recognition reliability g will become large if the estimated state value rises up through time and it will become low when the estimated state value goes down. Another condition is to keep g value in the range from 0 to 1.

2.3 Learning by Observation

In the previous section, behavior recognition system based on state value of its own behavior is described. This system shows robust recognition of observed behavior [5] only when the behavior to be recognized has been well-learned beforehand. If the behavior is under learning, then, the recognition system is not able to show good recognition performance at beginning. The trajectory of the observed behavior can be a bias for learning behavior and might enhance the behavior learning based on the trajectory. The observer cannot watch actions of observed behavior directly and can only estimate the sequence of the state of the observed robot. Let s_t^o be the estimated state of the observed robot at time t. Then, the estimated state value $\hat{V^o}$ of the observed behavior can be calculated as below:

$$\hat{V^o}(s) = \sum_{s'} \hat{\mathcal{P}}_{ss'}^o \left[\hat{\mathcal{R}}(s') + \gamma V^o(s') \right] \tag{6}$$

where $\hat{\mathcal{P}}_{ss'}^o$ is state transition probability estimated from the behavior observation. This state value function $\hat{V^o}$ can be used as a bias of the state value function of the learner V. The learner updates its state-action value function $Q(s,a)$ during trials and errors based on the estimated state value of observed behavior $\hat{V^o}$ as below:

$$Q(s,a) = \sum_{s'} \hat{\mathcal{P}}_{ss'}^a \left[\hat{\mathcal{R}}(s') + \gamma V'(s') \right] \tag{7}$$

while

$$V'(s) = \begin{cases} V(s) & \text{if } V(s) > \hat{V^o}(s) \text{ or the transition}(s,a) \to s' \text{is well experienced} \\ \hat{V^o}(s) & \text{else} \end{cases}$$

This is a normal update equation as shown in (4) except using $V'(s)$. The update system switches the state value of the next state s' between the state value of own learning behavior $V(s')$ and the one of the observed behavior $\hat{V^o}(s')$. It takes $V(s')$ if the state value of own learning behavior $V(s')$ is bigger than the one of the observed behavior $\hat{V^o}(s')$ or the state transition $(s,a) \to s'$ is well experienced by the learner, $\hat{V^o}(s')$ else. This means the state value update system takes $\hat{V^o}(s')$ if the learner does not estimate the state value $V(s')$ because of lack of experience at the state s' from which it reaches to the goal of the behavior. $\hat{V^o}(s')$ becomes a bias for reinforcing the action a from the state s even though the state value of its own behavior $V(s')$ is small so that it leads the learner to explore the space near to the goal state of the behavior effectively.

2.4 Modular Learning System

In order to observe/learn/execute a number of behavior in parallel, we prepare a number of behavior modules each of which adopts the behavior learning and behavior recognition method as shown in Fig. 5. The learner tries to acquire a number of behavior shown in Table 1. The table also describes necessary state variables for each behavior. A range of each state variable is divided into 11 in order to construct a quantized state space. 6 actions are prepared to be selected by the learning module for each behavior: Approaching the goal, approaching the teammate, approaching the ball, leaving from the ball, turn around the ball clockwise and counterclockwise.

A demonstrator is supposed to show a number of behavior which are not informed directly to the observer. In order to update the estimate values of the

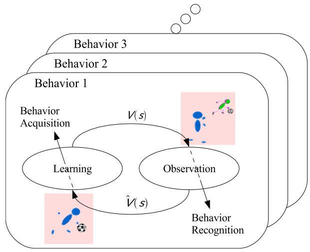


Fig. 5. Modular Learning System for Behavior Acquisition and Recognition

Table 1. List of behavior learned by self and state variables for each behavior

Behavior	State variables
Approaching a ball	d_b
Approaching a goal	d_g
Approaching the teammate	d_r
Shooting a ball	d_b, d_g, θ_{bg}
Passing a ball	d_b, d_r, θ_{br}

behavior the demonstrator is taking, the observer has to estimate which behavior the demonstrator is taking correctly. If the observer waits to learn some specific behavior by observation until it becomes able to recognize the observed behavior well, bootstrap of leaning unfamiliar behavior by observation cannot be expected. Therefore, the observer(learner) maintains a history of the observed trajectories and updates value functions of the observed behavior with

– high recognition reliability or
– high received reward.

The observer estimates the state of the demonstrator every step and the reward received by the demonstrator is estimated as well. If it is estimated that the demonstrator receives a positive reward by reaching to the goal state of the behavior, then, the observer updates the state value of the corresponding behavior even if it has low recognition reliability for the observed behavior. The update strategy enhances to estimate appropriate values of the observed behavior.

3 Behavior Learning by Observation

3.1 Experimental Setup

In order to validate the effect of interaction between acquisition and recognition of behavior through observation, two experiments are set up. One is that the learner does not observe the behavior of other but tries to acquire shooting/passing behavior by itself. The other is that the learner observes the behavior of other and enhances the learning of the behavior based on the estimated state value of the observed behavior. In the former experiment, the learner follows the learning procedure:

1. 15 episodes for behavior learning by itself
2. evaluation of self-behavior performance
3. evaluation of behavior recognition performance
4. goto 1.

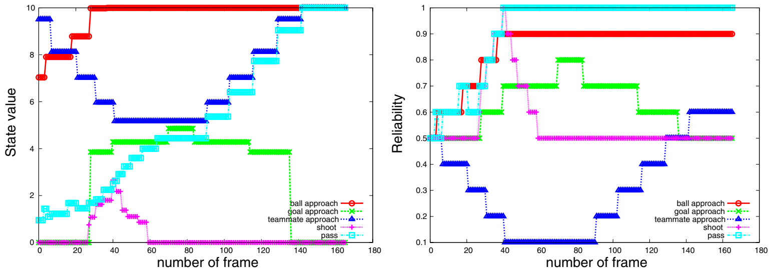
On the other hand, in the later experiment, it follows :

1. 5 episodes for observation of the behavior of the other
2. 10 episodes for behavior learning by self-trials with observed experience
3. evaluation of self-behavior performance
4. evaluation of behavior recognition performance
5. goto 1.

Both learners attempt to acquire behavior listed in Table 1. The demonstrator shows one of the behavior one by one but the observer does not know which behavior the demonstrator is taking. In both experiments, the learner follows ϵ-greedy method; it follows the greedy policy with 80% probability and takes a random action else. Performance of the behavior execution and recognition of observed behavior during the learning time is evaluated every 15 learning episodes. The performance of the behavior execution is the success rate of the behavior while the learner, the ball, and the teammate are placed at a set of pre-defined positions. The one of the behavior recognition is the average length of period in which the recognition reliability of the right behavior is larger than 70% during the observation. The soccer field area is divided 3 by 3 and the center of the each area is a candidate of the initial position of the ball, the learner, or the teammate. The performances are evaluated in all possible combinations of the positions.

3.2 Recognition of Observed Behavior

Before evaluating the performance of the behavior execution and behavior recognition of other during learning the behavior, we briefly review how this system estimates the values of behavior and recognizes the observed behavior after the observer has learned behavior. When the observer watches a behavior of the other, it recognizes the observed behavior based on repertoire of its own behavior. Figs. 6 (a) and (b) show sequences of estimated values and reliabilities of the behavior, respectively. The line that indicates the passing behavior keeps tendency of increasing value during the behavior in this figures. This behavior is composed of behavior of approaching a ball and approaching the teammate again, then, the line of approaching a ball goes up at the earlier stage and the line of approaching the teammate goes up at the later stage in Fig. 6(a). All reliabilities start from 0.5 and increase if the value goes up and decrease else. Even when the value stays low, if it is increasing with small value, the recognition reliability of the behavior increases rapidly. The recognition reliability of the



(a) Estimated Values (b) Reliabilities

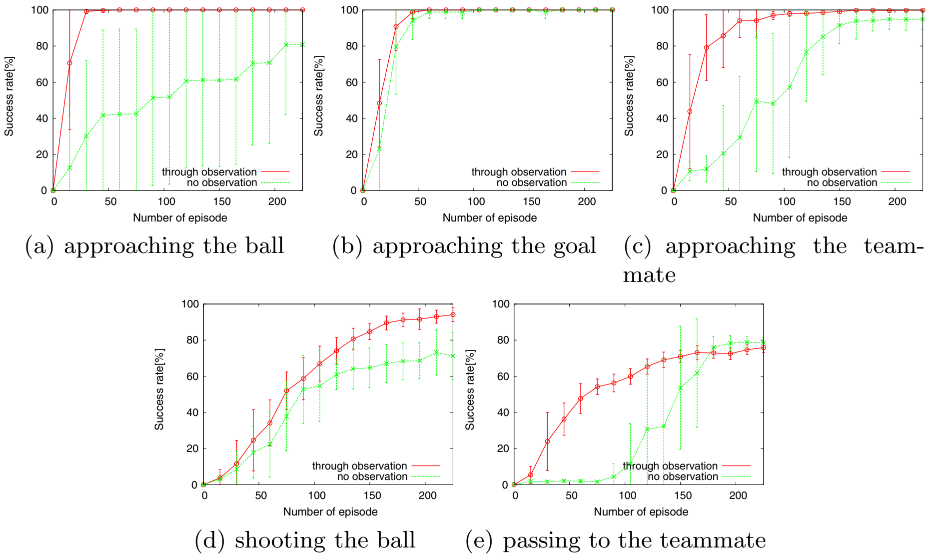
Fig. 6. Sequence of estimated values and reliabilities during a behavior of pushing a ball to the magenta player, red line : approaching a ball, green line : approaching the goal, light blue line : passing, blue line : approaching the other, magenta line : shooting

behavior of pushing a ball into the teammate reaches 1.0 at middle stage of the observed behavior. "Recognition period rate" of observed behavior is introduced here to evaluate how long the observer can recognize the observed behavior as a correct one. The recognition period rate is 85% here ,that means, the period in which the recognition reliability of the passing behavior is over 70% is 85% during the observation.

3.3 Performance of Behavior Learning and Recognition

In this section, performances of the behavior execution and behavior recognition during learning the behavior are shown. Fig. 7 shows success rates of the behavior and their variances during learning in cases of learning with/without value update through observation. The success rates with value update of all kinds of behavior grows more rapidly than the one without observation feedback. Rapid learning is one of the most important aspect for a real robot application. The success rate without observation sometimes could not reach the goal of the behavior at the beginning of the learning because there is no bias to lead the robot to learn appropriate actions. This is the reason why the variances of the rate is big. On the other hand, the system with value update through observation utilizes the observation to bootstrap the learning even though it cannot read exact actions of observed behavior.

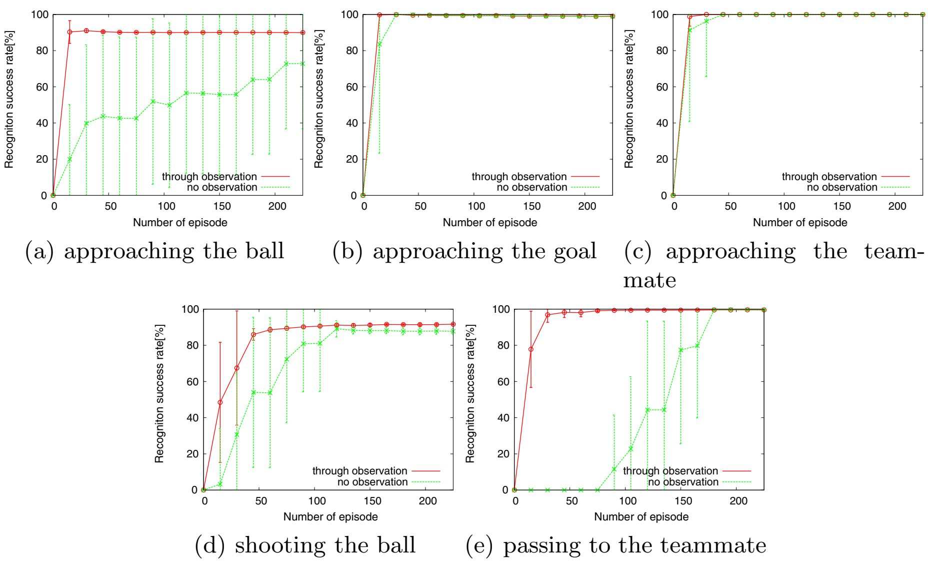
Recognition performance and recognition period rate of observed behavior and their variances are shown in Figs. 8 and 9, respectively. They indicate a similar aspect with the success rates of the acquired behavior. The performance of the behavior recognition depends on the learning performance. If the learning system has not experienced enough to estimate state value of the observed behavior, it cannot perform well. The learning system with value update with observed behavior rapidly enables to recognize the behavior while the system without value update based on the observation has to wait to realize a good recognition performance until it estimates good state value of the behavior by its own trials and errors.



(a) approaching the ball (b) approaching the goal (c) approaching the teammate

(d) shooting the ball (e) passing to the teammate

Fig. 7. Success rate of the behavior during learning with/without observation of demonstrator's behavior

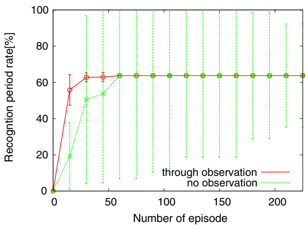


(a) approaching the ball (b) approaching the goal (c) approaching the teammate

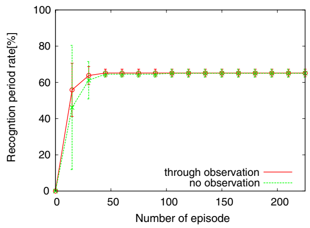
(d) shooting the ball (e) passing to the teammate

Fig. 8. Recognition performance of the behavior during learning with/without observation of demonstrator's behavior

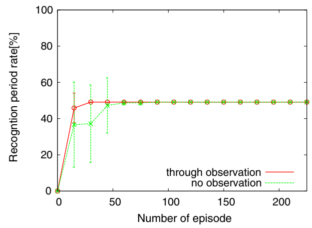
Those figures show the importance of learning through interaction between behavior acquisition and recognition of observed behavior.



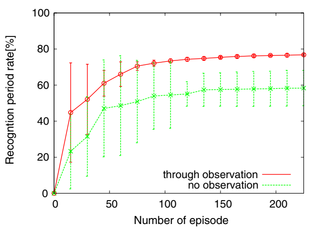
(a) approaching the ball



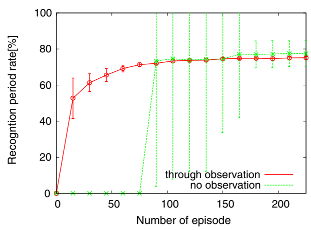
(b) approaching the goal



(c) approaching the teammate



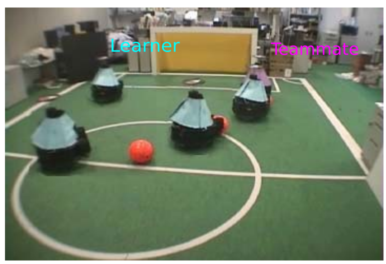
(d) shooting the ball



(e) passing to the teammate

Fig. 9. Recognition period rate of the behavior during learning with/without observation of demonstrator's behavior

(a) Demonstration by a human



(b) Learned behavior

Fig. 10. Demonstration of a passing behavior by a human and acquired passing behavior after the observation and learning

3.4 Experiments with a Real Robot and a Human Demonstrator

The proposed architecture was applied to the real robot system. The robot observed demonstrations of shooting and passing behavior played by a human player first, then, the robot learned the behavior by itself. The human player showed shooting and passing behavior 5 times for each. The real learning time was half an hour for each behavior. Figs. 10 (a) and (b) show one scene of human demonstration of the passing behavior and acquired passing behavior, respectively. The robot acquired the observed behavior within an hour.

4 Conclusion

The observer uses its own value functions to recognize what the demonstrator will do. Preliminary investigations in a similar context have been done by Takahashi et al. [5] and they showed better robustness of behavior recognition than a typical method. In paper, unknown behavior are also understood in term of own value function through learning based on the estimated values derived from the observed behavior. Furthermore, value update through the observation enhances not only the performance of behavior learning but also the one of recognition of the observed behavior effectively.

References

1. Bentivegna, D.C., Atkeson, C.G., Chenga, G.: Learning tasks from observation and practice. Robotics and Autonomous Systems 47, 163–169 (2004)
2. Connell, J.H., Mahadevan, S.: Robot Learning. Kluwer Academic Publishers, Dordrecht (1993)
3. Price, B., Boutilier, C.: Accelerating reinforcement learning through implicit imitatione. Journal of Articial Intelligence Research (2003)
4. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
5. Takahashi, Y., Kawamata, T., Asada, M.: Learning utility for behavior acquisition and intention inference of other agent. In: Proceedings of the 2006 IEEE/RSJ IROS 2006 Workshop on Multi-objective Robotics, pp. 25–31 (October 2006)
6. Takahashi, Y., Kawamata, T., Asada, M., Negrello, M.: Emulation and behavior understanding through shared values. In: Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3950–3955 (October 2007)
7. Whitehead, S.D.: Complexity and cooperation in q-learning. In: Proceedings Eighth International Workshop on Machine Learning (ML 1991), pp. 363–367 (1991)