

A Framework for an Interactive Robot-based Tutoring System and Its Application to Ball-passing Training

Dong-Hyun Lee and Jong-Hwan Kim

Abstract—This paper describes a framework for an interactive robot-based tutoring system (IRTS). The proposed IRTS is based on combining aspects of intelligent tutoring systems (ITSs) which are computer-based expert systems to simulate aspects of a human tutor, and robot-assisted systems which help users with physical interaction using robotic devices. The IRTS involves the robot interacting with a user, generating the user model, providing training tasks for guiding the user, and giving feedback on his/her performance. The proposed system is applied to ball-passing training. In the training, a human trainee passes a ball to a robot tutor and receives feedback from the robot on how to improve the kick, based on the angle and velocity of the ball. In the experiment, the system is implemented in a real robot and demonstrates that the system is able to provide user-adaptive training tasks and feedback to improve his/her ball-passing skill.

I. INTRODUCTION

Using robots for human learning is based on a concept that humans learn knowledge and skill from robots. In recent years, the concept is applied to the area of education such that kindergartens and elementary schools have been adopting the robot-based learning system at class [1],[2]. Since robots can acoustically, visually, and physically interact with people, they can provide various types of educational contents or training programs to the users with using gestures, expressions, and other cues. The conventional approach for physical training requires assistance of a skilled trainer to facilitate learning, which is expensive and labor-intensive. As a result, people who do not learn quickly may experience limited training time, preventing them from achieving independent learning ability [3]. To lower the cost and improve accessibility to training, a robot-based ball passing training system has developed where people can safely develop their passing skills at their own pace with minimum assistance from a trainee. Robot-based training is not only more efficient in training repetitive physical activities, but also provides a model of the trainee, customization of the training, and maintenance of trainee records [4].

There are mainly two types of systems that can teach and train human learners, intelligent tutoring systems (ITSs) and robot-assisted systems. The intelligent tutoring systems are any computer systems that provide direct customized instruction or feedback to students without the intervention of human beings [5]. There have been various approaches to improve the teaching ability of the ITS. Sklar et al. built educational keyboarding games as an ITS for educating children [6],[7]. To supply contents for the game in a web-based learning environment, an evolutionary method was applied. Hernandez et al. proposed an ITS with combining

the affective and cognitive state of the student to establish the affective and pedagogical actions [8]. However, since most of the ITSs are based on the human-computer interaction such that the students interact with the system through a graphical user interface and a simulation of the task domain, they are not able to supply the teaching programs which require physical development of the students. Therefore, the conventional ITSs might be inappropriate for teaching them skilled motions and sports in a real environment.

The robot-aided training systems have been applied in the field of rehabilitation such as training arms, ankles and leg movements of patients [9]-[12]. They have also been used for training human hand impedance and improving surgical skills [13],[14]. To improve the motor learning experience of the patients, various training algorithms have been proposed. Reinkensmeyer et al. examined the effect of robotic assistance on human motor learning [15]. Patton et al. tested the possibility of teaching desired movements to the patients by using robot-applied force fields [16]. However, most of the researches have focused on assisting human motor learning for adjusting muscle activations. Since these algorithms are based on neuromuscular and biomechanical characteristics of the human body, they are not applicable for teaching or training the users a general educational contents.

To deal with these issues, this paper proposes a framework for an interactive robot-based tutoring system (IRTS). In the framework, a robot tutor interacts with a user, measures the user performance by using its sensors during the training, generates the user model, and provides appropriate training tasks and training feedbacks to the user based on his/her proficiency and propensity. The training task consists of the proficiency-based and propensity-based training units. The proficiency-based training unit provides the training program which is suitable for the ability of the user. The propensity-based training unit helps the user make up for his/her habitual weakness. For the training feedback, the rule-based and score-based training feedbacks are introduced. The rule-based training feedback, which is useful for a novice, provides the expert's opinion on the user's performance by using a rule-based expert system. The score-based training feedback, which is useful for a semi-expert, gives a score to the user based on the user's performance.

The rest of this paper is organized as follows. In Section II, the details of the proposed framework is presented. Section III describes the implementation of the framework on the ball passing training. The experimental results are discussed in Section IV and concluding remarks follow in Section V.

II. INTERACTIVE ROBOT-BASED TUTORING SYSTEM

Fig. 1 shows the proposed interactive robot-based tutoring system architecture. As shown in the figure, the training

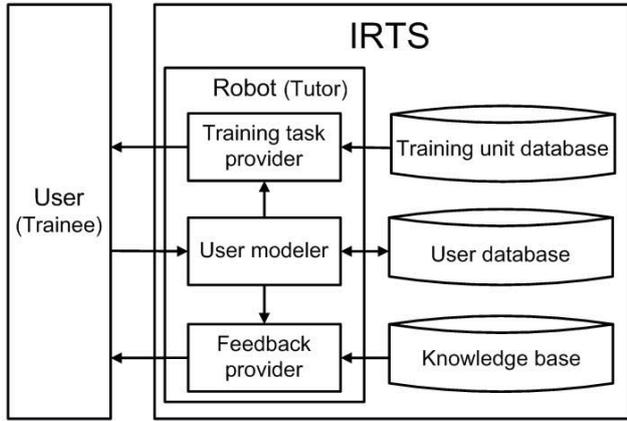


Fig. 1. The architecture of IRTS.

task is generated from the training task provider, and the user performs the training task by interacting with the robot. The training task is regarded as a training instruction which can help the user improve his/her skill. It consists of the proficiency-based and propensity-based training units. The proficiency-based training unit provides a training instruction to the user with a suitable degree of difficulty considering the proficiency of the user. By using this, the system is able to prevent the user from discouragement or losing interest during the training. The propensity-based training unit is used to make up for the habitual weakness or defect of the user. For example, suppose that a user learns how to play pingpong with a robot trainer. During the training, the robot analyzes the performance of the user and recognizes that the user does not receive the ball well when it comes to the left-side of the user. The robot then hits the ball toward the left-side of the user in the next trial to correct user's left-side weakness.

While the user performs the training tasks, the robot measures not only the physical movement of the user but also the result of the user's movement. For example, if the user learns how to shoot a basketball, the physical movement of the user such as the movement of user's knees and elbows, and the result of user's movement such as the angle between the rim and the thrown ball are considered as the performance of the user. After measuring the user performance, the user modeler generates the user model which consists of his/her proficiency and propensity. If there already exists a record about the user in the user database, the user model is fetched from the user database, and if not, a new user model is created and recorded to the user database. The user model is continuously updated during the training.

Based on the user model, the feedback provider selects one of the rule-based and score-based training feedbacks.

The rule-based training feedback provides basic guidelines for the user to improve his/her skill by using a rule-based expert system. A rule in the system is stored as an IF-THEN structure where the IF part represents the performance of the user, and the THEN part shows the expert's knowledge for guiding the user to improve his/her performance. For example, if the user learns a foreign language and has difficulty in pronouncing some characters, the robot measures the movement of the tongue and the pronounced sound of the user, and uses the measurement data as the IF part of the rule. Then the expert system fires the rule by providing the THEN part, such as "Push the tip of the tongue to the gum ridge much harder." This type of training feedback is useful for novices since they do not know how to perform the task correctly, and how to improve their skill. The score-based training feedback gives a score to the user about his/her performance. The user gets higher scores as his/her performance is closer to the target performance. It is useful for semi-experts since they already have some degree of knowledge about how to improve their skill, and how to perform correctly, but do not get accustomed yet. It is also useful for them to develop their own ideas or methods without having to stick to fixed instructions.

The selection of the training feedback is done stochastically considering the proficiency of the user. If the proficiency of the user is low, the probability for selecting the rule-based training feedback increases. As the user performs better during the training, the probability of selecting the score-based training feedback increases. After selecting the training feedback, the robot provides the selected training feedback to the user in the form of synthesized sound, visual display, as well as the physical feedback by using its mechanical device. For example, if the user learns how to play bowling by using the exoskeleton-type robot which is attached to the arm of the user, the robot provides the physical feedback to the user such as holding and releasing his/her elbow joint to correct the posture movement.

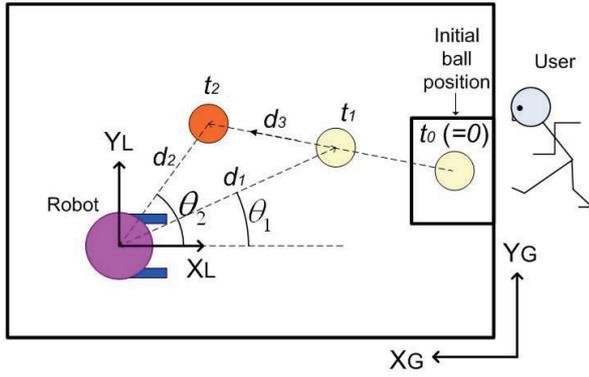
III. BALL-PASSING TRAINING

The proposed framework was applied to the ball passing training. To pass a ball properly, the user should kick the ball with accurate angle and moderate speed such that the receiver can catch the ball easily. Fig. 2 shows an overview of the ball-passing training. Fig. 2 (a) shows the real environment of the ball-passing training. The robot has an omni-directional vision system and omni-directional mobility such that it can measure the angle and distance of the approaching ball, and track the ball without nonholonomic constraints [17]. Fig. 2 (b) shows the angle and distance of the ball from the robot, where $\theta_1(\theta_2)$ and $d_1(d_2)$ represent the angle and distance of the ball based on the origin of the robot local reference frame $\{X_L, Y_L\}$ at time $t_1(t_2)$, respectively. To calculate the angle and velocity of the ball, the robot measures the angle and distance of the ball twice at t_1 and t_2 . Then the distance between the ball at t_1 and t_2 , d_3 is calculated as

$$d_3 = \sqrt{d_1^2 + d_2^2 - 2d_1d_2 \cos(\theta_1 - \theta_2)}. \quad (1)$$



(a)



(b)

Fig. 2. An overview of the ball-passing training.

Then the angle and the velocity of the ball, θ_B and v_B are defined as

$$\theta_B = \theta_2, \quad v_B = d_3 / (t_2 - t_1). \quad (2)$$

Since the robot starts tracking the ball after t_2 , the velocity of the robot is not considered in (1) and (2).

The scenario of the training is described as follows:

- 1) The robot takes its position based on the training task and waits until the user passes the ball.
- 2) The user passes the ball to the robot.
- 3) The robot receives the ball and returns the ball back to the user.
- 4) The robot provides the user with the training feedback of the trial.
- 5) Repeat from 1) to 4) until the episode is finished.
- 6) The robot provides the user with the training feedback of the episode.
- 7) Repeat from 1) to 6) until all of the episodes are finished or termination conditions are met.

The ball-passing training consists of episodes and trials where one episode consists of a set of trials. For each trial the user passes the ball to the robot, and the robot receives the

ball and sends it back to the user. During the trials, the system records the user performance data, and when one episode is finished, the recorded data during the trials are utilized for generating the user model. The system then generates the training task for the next episode based on the user model.

In order to determine the proficiency and propensity of the user, the system requires the target performance vector, Y , which is defined as

$$Y = [\theta_T, v_T] \quad (3)$$

where θ_T and v_T are the target angle and the target velocity of the ball, respectively. In the experiment, Y was set to $[0.0 \text{ (rad)}, 0.7 \text{ (m/s)}]$. The user performance vector of the i th trial, $X(i)$ is defined as

$$X(i) = [\theta_B(i), v_B(i)] \quad (4)$$

where $\theta_B(i)$ and $v_B(i)$ are the angle and velocity of the ball at the i th trial, respectively. The performance error of the i th trial is then defined as

$$E(i) = [e_1(i), e_2(i)] \quad (5)$$

where

$$e_1(i) = \theta_T - \theta_B(i), \quad e_2(i) = v_T - v_B(i). \quad (6)$$

The performance of the user for each trial is represented as a score and tendency. The score indicates how well the user performs the task, and the tendency represents the propensity of the user during the trial. To obtain the score and tendency, the scaled error vector of the i th trial, $\hat{E}(i)$ is defined as

$$\hat{E}(i) = [\hat{e}_1(i), \hat{e}_2(i)] \quad (7)$$

with

$$\hat{e}_1(i) = e_1(i) / \Theta, \quad \hat{e}_2(i) = e_2(i) / V \quad (8)$$

where Θ and V are the maximum allowable angle error and velocity error, respectively. In the experiment, Θ and V were set to 0.3 (rad) and 0.3 (m/s) , respectively. The scaled error vector is in the two-dimensional space as shown in Fig. 3. The score for each trial is considered as a short-term proficiency of the user. It is obtained by the range of the magnitude of the scaled error vector as 100 points in $0.0 \sim 0.3$, 60 points in $0.3 \sim 0.6$, and 30 points in $0.6 \sim 1.0$. If the magnitude is beyond 1.0, the user gets 0 point. The tendency for each trial represents a short-term propensity of the user. To define the tendency of the user, \hat{e}_1 and \hat{e}_2 are divided into seven classes, respectively, as shown in Table I. For example, if the scaled error vector is the point a $(-1.2, -0.8)$ in Fig. 3, then the score is 0 and the tendency of the user is VR and S . Likewise, if the vector is the point b $(0.5, 0.5)$, then the score is 30 and the tendency is LL and LF .

After finishing all of the trials in the episode, the proficiency and propensity of the user are defined. The user proficiency of the j th episode, $U_S(j)$ is defined as

$$U_S(j) = \begin{cases} \text{Novice} & \text{if } 0 \leq Q(j) < 30 \cdot N \\ \text{Semi-expert} & \text{if } 30 \cdot N \leq Q(j) < 60 \cdot N \\ \text{Expert} & \text{otherwise} \end{cases} \quad (9)$$

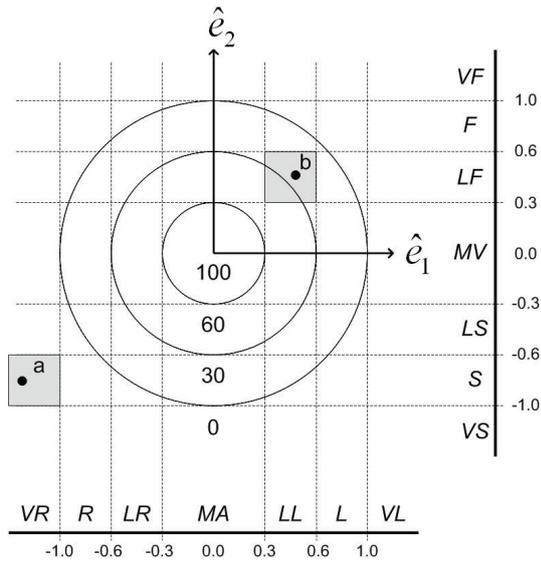


Fig. 3. The score and the tendency of the user in the scaled error vector space.

TABLE I
CLASS OF THE USER TENDENCY

Angle	Description	Velocity	Description
<i>VR</i>	Very right-side	<i>VF</i>	Very fast
<i>R</i>	Right-side	<i>F</i>	Fast
<i>LR</i>	Little right-side	<i>LF</i>	Little fast
<i>MA</i>	Moderate angle	<i>MV</i>	Moderate velocity
<i>LL</i>	Little left-side	<i>LS</i>	Little slow
<i>L</i>	Left-side	<i>S</i>	Slow
<i>VL</i>	Very left-side	<i>VS</i>	Very slow
<i>NOA</i>	No angle propensity	<i>NOV</i>	No velocity propensity

where $Q(j)$ represents the total sum of the scores in the j th episode, and N is the number of trials in the episode. In the experiment, N was set to 10.

The propensity of the j th episode, $U_T(j)$ is defined as

$$U_T(j) = [T_1, T_2] \quad (10)$$

where $T_1 \in \{VR, R, LR, MA, LL, L, VL\}$ is the angle propensity, and $T_2 \in \{VF, F, LF, MS, LS, S, VS\}$ is the velocity propensity of the user, respectively. The user propensity class is selected considering the number of scaled error vectors in each of the classes. If there exists a dominant class which has considerably larger number of scaled error vectors than any other classes, it is selected as the propensity of the user. If there is no dominant class due to the widely dispersed scaled error vectors, *NOA* and *NOV* will be selected for the angle class and the velocity class, respectively.

For stochastic training feedback selection, a Bernoulli distribution is used to define the feedback selection probability [18]. The feedback selection probability of the j th episode, $p^S(j)$ is defined as

$$p^S(j) = c + \frac{1 - c}{1 + \exp(-a \cdot (Q(j) - b))} \quad (11)$$

where a , b and c respectively determine the slope, the center of the slope, and the lowest probability. In the experiment,

a , b and c were set to 0.1, 500 and 0.2, respectively. $p^S(j)$ is determined at the end of the j th episode, and it is used as the probability value of selecting the score-based training feedback $((1 - p^S(j))$ for the rule-based training feedback) at $(j + 1)$ th episode. As the total sum of the scores increases during the sequence of the episodes, the feedback selection probability is increased accordingly. As a result, if the total sum of the scores of the user is large in the current episode, the user is likely to get the score-based training feedback more than the rule-based training feedback in the next episode.

The training task provides the training instruction to the user by changing the initial position of the robot, R_{POS} in the global reference frame $\{X_G, Y_G\}$, which is define as

$$R_{POS} = [x_R, y_R]. \quad (12)$$

The proficiency-based training unit changes x_R which represents the distance between the robot and the user as described in Table II. By changing the distance based on the user

TABLE II
DESCRIPTION OF THE PROFICIENCY-BASED TRAINING UNIT

Proficiency	x_R (m)
<i>Novice</i>	2.0
<i>Semi-expert</i>	3.5
<i>Expert</i>	5.0

proficiency, the proficiency-based training unit can adjust the degree of difficulty of the training level. In the experiment, y_R was set to 2.5 m. The propensity-based training unit modifies R_{POS} by augmenting additional values (η and ζ) as shown in Table III. In the table, $rand(-1.0, 1.0)$ returns

TABLE III
DESCRIPTION OF THE PROPENSITY-BASED TRAINING UNIT

Propensity	ζ (m)	Propensity	η (m)
<i>VR</i>	-1.0	<i>VF</i>	-1.0
<i>R</i>	-0.6	<i>F</i>	-0.6
<i>LR</i>	-0.3	<i>LF</i>	-0.3
<i>MA</i>	$rand(-1.0, 1.0)$	<i>MV</i>	$rand(-1.0, 1.0)$
<i>LL</i>	0.3	<i>LS</i>	0.3
<i>L</i>	0.6	<i>S</i>	0.6
<i>VL</i>	1.0	<i>VS</i>	1.0
<i>NOA</i>	0.0	<i>NOV</i>	0.0

a random value between -1.0 and 1.0 . By combining the proficiency-based and propensity-based training units, the initial position of the robot is defined as

$$R_{POS} = [x_R + \eta, y_R + \zeta]. \quad (13)$$

For example, if the user proficiency of the current episode is *Semi-expert* and the user propensity is $U_T = [R, LS]$, ζ and η are set to -0.6 and 0.3 , respectively. Then the initial position of the robot for the next episode is defined as $R_{POS} = [3.5 + 0.3, 2.5 - 0.6] = [3.8, 1.9]$.

The user propensity is used as an IF part of the rule-based training feedback. The rules for the angle and velocity of the ball are described in Table IV and V, respectively.

TABLE IV
RULES FOR ANGLE CORRECTION

Rule ^A	IF	THEN
Rule ₁ ^A	VR	Kick more to the left
Rule ₂ ^A	R	Kick to the left
Rule ₃ ^A	LR	Kick a little bit to the left
Rule ₄ ^A	MA	Maintain the direction
Rule ₅ ^A	LL	Kick a little bit to the right
Rule ₆ ^A	L	Kick to the right
Rule ₇ ^A	VL	Kick more to the right
Rule ₈ ^A	NOA	Kick straight to the robot

TABLE V
RULES FOR VELOCITY CORRECTION

Rule ^V	IF	THEN
Rule ₁ ^V	VF	Kick much slower
Rule ₂ ^V	F	Kick slower
Rule ₃ ^V	LF	Kick a little bit slower
Rule ₄ ^V	MV	Maintain the speed
Rule ₅ ^V	LS	Kick a little bit faster
Rule ₆ ^V	S	Kick faster
Rule ₇ ^V	VS	Kick much faster
Rule ₈ ^V	NOV	Kick with gentle speed

IV. EXPERIMENTAL RESULTS

In the experiment, four episodes were performed by the user where each episode consisted of ten trials. Before starting the first episode, the user informed that he was between a novice and a semi-expert for ball-passing so that the initial position of the robot was set to $R_{POS} = [3.5, 2.5]$. Fig. 4 shows the result of the scaled error vectors for each episode, and Fig. 5 shows the change of the initial position of the robot and the user model (user proficiency and user propensity) during the episodes. In Fig. 4, '*' represents the scaled error vector for each of the trials. In the first episode, the total score of the user was 220 points so that the user proficiency was defined as $U_S(1) = \text{Novice}$. Since the scaled error vectors were widely dispersed as shown in Fig. 4, the user propensity was defined as $U_T(1) = [NOA, NOV]$. These results represented that the user was not good at ball passing, and he did not have any kind of specific propensity during the episode. Since the proficiency of the user in the first episode was *Novice*, x_R was set to 2.0 by the proficiency-based training unit. As a result, the initial position of the robot for the next episode was set to $R_{POS} = [2.0, 2.5]$. In the second episode, the total score was 200 points, and the user showed the propensity of passing the ball with low velocity. To make the user kick the ball faster, the system provided the initial position of the robot for the next episode as $R_{POS} = [2.6, 2.5]$ (η was set to 0.6 and ζ was set to 0 as in Table III). In the third episode, the total score was 430 points, and the user tended to pass the ball with moderate angle and fast speed. Since the user proficiency was $U_S(3) = \text{Semi-expert}$, x_R was set to 3.5 m to give the user a more challenging task. To correct user's habitual weakness of passing the ball fast, η was set to -0.6 m (as in Table III). Since the user was good at passing the ball

with moderate angle, ζ was selected randomly between -1.0 and 1.0 in order to reinforce user's skill. In the experiment, the randomly selected value for ζ was -0.4 . As a result, the initial position of the robot for the next episode was set to $R_{POS} = [3.5 - 0.6, 2.5 - 0.4] = [2.9, 2.1]$. In the last episode, the user proficiency was $U_S(4) = \text{Expert}$ with the total score of 770 points, and the user propensity was $U_T(4) = [MA, MV]$. As shown in Fig. 5, the user gradually

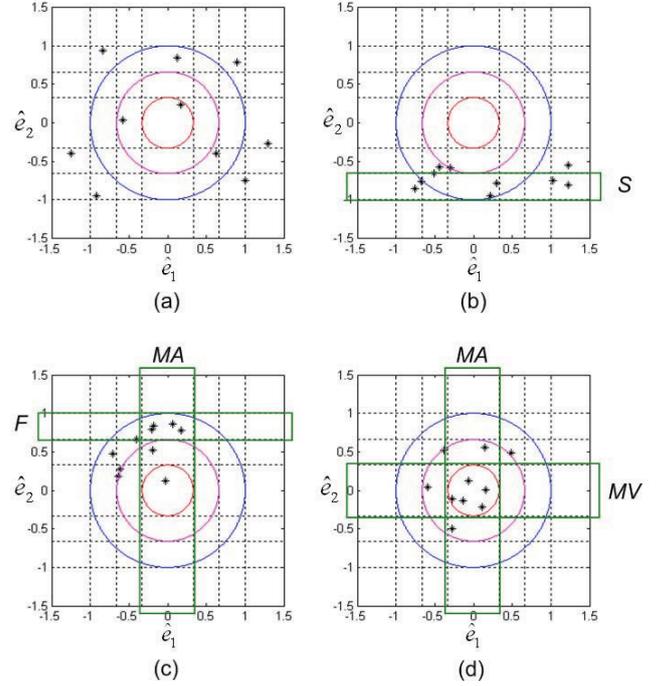


Fig. 4. The results of the episodes (a)1st episode, (b)2nd episode, (c)3rd episode, (d)4th episode.

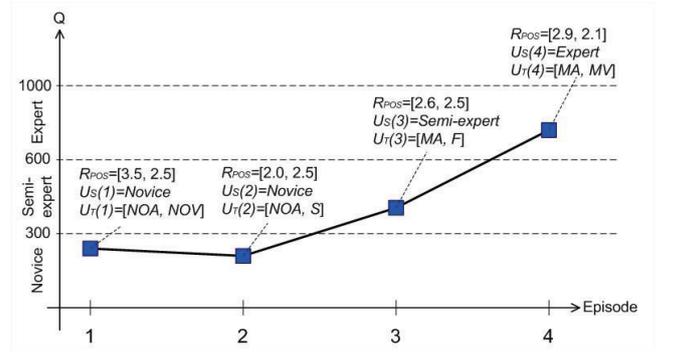


Fig. 5. Change of the task performance and user model during the sequence of the episodes.

increased the total score and rectified the habitual weakness during the episodes.

V. CONCLUSIONS

This paper proposed a robot-based learning framework for the interactive robot-based tutoring system (IRTS). In this framework, a robot tutor interacted with a user in

a real environment, and trained the user considering his ability and habitual weakness. Since the system generated the proficiency and propensity of the user during the training, the system was able to provide a suitable training task to the user by combining the proficiency-based and propensity-based training units. The training task was provided to the user by taking advantage of the robot's capability such as mobility, manipulation, voice synthesizing. To give a suitable feedback to the user, either the rule-based or score-based training feedback was selected stochastically considering the user's proficiency. The system was applied to the ball-passing training and the experiment result demonstrated that the system was able to improve user's ability as well as correcting the habitual weakness.

As a future work, the experiments will be expanded to various types of subjects with different ages and abilities to validate the effectiveness of the system.

ACKNOWLEDGMENT

This work was supported (National Robotics Research Center for Robot Intelligence Technology, KAIST) by Ministry of Knowledge Economy under Human Resources Development Program for Convergence Robot Specialists.

REFERENCES

- [1] T. Kanda, T. Hirano, D. Eaton, and H. Ishiguro, "Interactive Robots as Social Partners and Peer Tutors for Children: a field trial," *Human-Computer Interaction J.*, vol. 19, pp. 61-84, 2004.
- [2] K. Oh and M. Kim, "Social Attributes of Robotic Products: Observations of Child-Robot Interactions in a School Environment," *Design. J.*, vol. 4, pp. 45-55, 2010.
- [3] L. M. Crespo, J. Furumasu, D. J. Reinkensmeyer, "A robotic wheelchair trainer: design overview and a feasibility study," *Neuroeng. Rehab. J.*, vol. 7, pp. 1-12, 2010.
- [4] H. I. Krebs, J. J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Ranneklev, B. T. Volpe and N. Hogan, "Rehabilitation robotics: Performance-based progressive robot-assisted therapy," *Auton. Robot.*, vol. 15, pp. 7-20, 2003.
- [5] J. Psocka, L. D. Massey, and S. A. Mutter, "Intelligent Tutoring Systems: Lessons Learned," Hillsdale, Lawrence Erlbaum, pp. 409-434, 1988.
- [6] E. Sklar, A. D. Blair, and J. Pollack, "Co-evolutionary Learning: Machines and Humans Schooling Together," in *World Cong. Expert Syst.*, 1998.
- [7] E. Sklar and J. Pollack, "An Evolutionary Approach to Guiding Students in an Educational Game," in *Proc. Int. Conf. Simulation Adapt. Beh.*, 2000.
- [8] Y. Hernandez, J. Noguez, E. Sucar, and G. A.-Figueroa, "A Probabilistic Model of Affective Behavior for Intelligent Tutoring Systems," *Lec. Note. Comp. Sci.*, vol. 3789, pp. 1175-1184, 2005.
- [9] Y. Tanaka, M. Ishii, T. Tsuji, and N. Imamura, "Modeling and Evaluation of Human Motor Skills in a Virtual Tennis Task," in *Proc. IEEE Int. Conf. Engin. Medicin. Bio. Soc.*, pp. 4190-4193, 2008.
- [10] D. Feygin, M. Keehner, and F. Tendick, "Haptic Guidance: Experimental Evaluation of a Haptic Training Method for a Perceptual Motor Skill," in *Proc. Int. Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst.*, pp. 40-47, 2002.
- [11] J. L. Emken and D.J. Reinkensmeyer, "Robot-enhanced Motor Learning: Accelerating Internal Model Formation During Locomotion by Transient Dynamic Amplification," *IEEE Trans. Neural Syst. Rehab. Engin.* vol. 13, pp. 33-39, 2005.
- [12] A. Roy, H. I. Krebs, D. J. Williams, C. T. Bever, L. W. Forrester, R. M. Macko, and N. Hogan, "Robot-aided Neurorehabilitation: A Novel Robot for Ankle Rehabilitation," *IEEE Trans. Rob.* vol. 25, pp. 569-582, 2009.
- [13] Y. Takeda, Y. Tanaka, and T. Tsuji, "A Virtual Training Sports System for Measuring Human Hand Impedance," in *Proc. IEEE/ASME Int. Conf. Advanced Intelli. Mech.*, pp. 914-919, 2003.
- [14] R. Sarle, A. Tewari, A. Shrivastava, J. Peabody, and M. Menon, "Surgical Robotics and Laparoscopic Training Drills," *Endourol. J.*, vol. 18, pp. 63-66, 2004.
- [15] J. L. Patton and D. J. Reinkensmeyer, "Can Robots Help the Learning of Skilled Actions?," *Exerc. Sport Sci. Rev.* vol. 37, pp. 43-51, 2009.
- [16] J. L. Patton, F. A. Mussa-Ivaldi, "Robot-assisted Adaptive Training: Custom Force Fields for Teaching Movement Patterns," *IEEE Trans. Biomed. Eng.*, vol. 51, pp. 636-46, 2004.
- [17] D.-H. Lee, J.-S. Sun, S.-B. Han, C.-S. Park and J.-H. Kim, "Omni-directional Robot System and Localization for FIRA ROBOSOT," in *Proc. FIRA Robot World Congress*, 2008.
- [18] J. Johns, S. Mahadevan, and B. Woolf, "Estimating Student Proficiency Using an Item Response Theory Model," in *Proc. Int. Conf. Intelli. Tutor. Syst.*, 2006.