Teaching a Pet Robot through Virtual Games

Anja Austermann[1] and Seiji Yamada[1,2]

[1] The Graduate University for Advanced Studies, Sokendai
[2] National Institute of Informatics
101-8430 Tokyo, Japan
{anja,seiji}@nii.ac.jp

Abstract. In this paper, we present a human-robot teaching framework that uses "virtual" games as a means for adapting a robot to its user through natural interaction in a controlled environment. We present an experimental study in which participants instruct an AIBO pet robot while playing different games together on a computer generated playfield. By playing the games in cooperation with its user, the robot learns to understand the user's natural way of giving multimodal positive and negative feedback. The games are designed in a way that the robot can reliably anticipate positive or negative feedback based on the game state and freely explore its user's reward behavior by making good or bad moves. We implemented a two-staged learning method combining Hidden Markov Models and a mathematical model of classical conditioning to learn how to discriminate between positive and negative feedback. After finishing the training the system was able to recognize positive and negative reward based on speech and touch with an average accuracy of 90.33%.

1 Introduction

In recent years, a lot of research has been done focusing on creating robots that are able to communicate with humans and learn from humans in a natural way. When teaching a robot in a natural environment, many issues have to be handled that are not directly related to the interaction with a human, but to perceiving and modeling the environment as well as moving around and manipulating objects. Even apparently simple tasks like picking up objects cause considerable implementation effort.

Using a robot simulation or a virtual agent can be an alternative in many cases but has the disadvantage that interaction cannot be perceived through the actual sensors of the robot and does not occur in the same spatial context as with a real robot. Moreover, especially in case of gesture or touch, user behavior depends on inherent properties of the robot like its size and the location of its sensors and can be expected to differ significantly between interacting with a real robot and a computer simulation. Therefore we implemented a client-server based framework for teaching a real robot in a "virtual" task, that is, a computer-generated visual representation of a task, where all relevant information can

H. Prendinger, J. Lester, and M. Ishizuka (Eds.): IVA 2008, LNAI 5208, pp. 308–321, 2008.

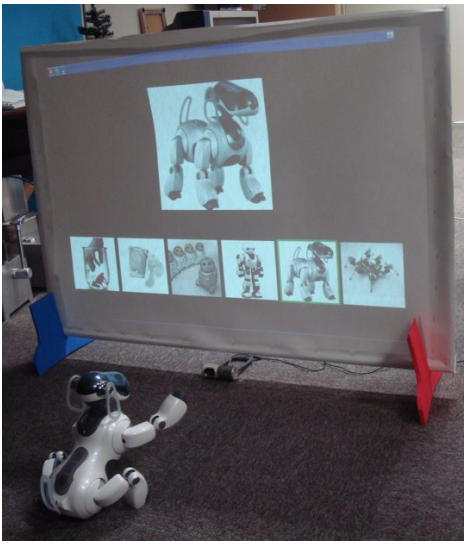

Fig. 1. AIBO during task execution

be accessed and controlled directly without additional effort for implementing perception and physical manipulation of the environment.

We present an experimental study that uses "virtual" games to allow an AIBO pet robot to learn to understand multimodal positive and negative feedback from a human through natural interaction. The setting is shown in figure 1. The use of virtual games allow us to create a controlled environment in which the robot can deliberately provoke and explore its user's reward behavior by making good or bad moves. Being able to instantly assess the correctness of a move, the robot can anticipate positive or negative reward and learn its user's preferred methods for giving feedback. The game tasks are explained in detail in section 4.2.

We chose understanding reward as a first step toward learning more general commands, because understanding whether an action has been correct or incorrect through human feedback is one of the capabilities that a robot usually needs when learning through interaction with a human instructor. In most existing service- or entertainment robot platforms, the means of giving reward to a robot are hard-coded such as predefined commands, buttons that have to be pressed or GUI-items of a remote application that have to be used for input. The user has to read a handbook and remember the correct way of giving commands and feedback. In order to enhance the user experience and to make interacting with a robot more accessible e.g. for aged people with memory deficits it would be desirable to shift the effort of learning and remembering the correct way of interacting from the user to the robot.

We propose a two-staged learning method for adapting the robot to its user's feedback. In the first stage Hidden Markov Models are used to learn to discriminate different perceptions in an unsupervised way. Then associations are learned

between perceptions represented by their corresponding HMMs and either positive or negative reward based on a mathematical model of classical conditioning. Details of the learning method is are given in section 5.

We conducted an experimental study in order to assess how humans give feedback to a robot in a virtual game task and analyzed the observed reward behaviors. We found that the two most important modalities for giving rewards are speech and touch, while gestures were mainly used for giving instructions, not reward. We also asked the users to answer a questionnaire about their experience during the experiments to find out which features of a training task are important for successful and enjoyable teaching. With our learning method an average recognition accuracy of 90.33% is reached for discriminating between positive and negative reward based on speech and touch.

2 Related Work

Approaches to combine actual robots with virtual or mixed reality have mainly been researched upon in the field of telerobotics. However, due to the distance between the robot, and the user, the modalities used for interaction typically differ from the ones used in face-to-face communication. The most closely related work from the field of telerobotics was developed by Xin and Sharlin [11]. They are using a mixed-reality implementation of the classic Sheep and Wolves game. The sheep is a virtual, computer generated object and has to be chased by a team of four robotic wolves on a real playfield. The human is part of the robot team and interacts with the robots. The user does not have direct contact with the robot but observes the playfield through an online mixed-reality system showing the current situation on the playfield. However, interaction is not done by physically interacting with the robots but from a distance through a text-based interface. Our work focuses on modalities that are naturally used when interacting with a robot in close distance, such as speech and touch.

Another related research field is the acquisition of speech and especially the grounding of vocabulary [5] [6] through human-robot interaction.

Steels and Kaplan [10] developed a system to teach the names of three different objects to an AIBO pet robot. They used so-called "language games" for teaching the connection between visual perceptions of an object and the name of the object to a robot through social learning with a human instructor.

Iwahashi described an approach [6] to the active and unsupervised acquisition of new words for the multimodal interface of a robot. He applies Hidden Markov Models to learn verbal representations of objects, perceived by a stereo camera. The learning component uses pre-trained HMMs as a basis for learning and interacts with its user in order to avoid and resolve misunderstandings.

Kayikci et al. [8] use Hidden Markov Models and a neural associative memory for learning to understand short speech commands in a three-staged recognition procedure. First, the system recognizes a speech signal as a sequence of diphones or triphones. In the next step, the sequences are translated into words using a

neural associative memory. The last step employs a neural associative memory to finally obtain a semantic representation of the utterance.

In the same way as the approaches outlined above, our learning algorithm attempts at assigning meanings to observations. However, our system is not trying to learn the relationship of individual words or symbols to real-world objects but focuses on relating observations to the concepts of positive or negative feedback. Those observations can be words as in the studies above, but also touch patterns, utterances consisting of multiple words and combinations of them. Moreover, our proposed approach is not limited to a single modality but tries to integrate observations from different modalities.

For learning associations between the meaning of commands and rewards and their appropriate Hidden Markov Model representations, classical conditioning is used. Mathematical theories of classical conditioning were extensively researched upon in the field of cognitive psychology. An overview can be found in [4].

3 Framework Design and Implementation

The focus of the actual implementation of the system was to develop a framework for conducting experiments that is easy to extend and to adapt to new tasks. It is implemented using a client-server based architecture consisting of four components which communicate via TCP/IP:

- *The game server* provides the display and handling of the playfield, an evaluation function for the robot's moves as well as the opponent's artificial intelligence in case of a game for multiple players.
- *The perception server* records and processes audio and video data of the user's interaction. It receives data from the robot's touch sensors, video data from two Logitech Fusion web cameras as well as audio data from a wireless lavalier microphone that is attached to the user's clothes. The data from different modalities is synchronized and stored, while the information, which is extracted from the audio and video data streams is sent to the robot control software. Learning to interpret the user's behavior using the method described in section 5 takes place in the perception server
- *The robot control software* is connected to the game server as well as the perception server and uses information about the game state to calculate the next moves of the robot. Moreover, it uses information from the perception server in order to assess whether interaction has been perceived in order to react appropriately.
- *The AIBO robot itself.* We are using an AIBO ERS-7 for our experiments. The AIBO Remote Framework [1] is used by the robot control software for wireless control of the robot and for reading its sensor data.

4 The Training Tasks

During the experiments, the image of the playfield is generated by a computer and projected from the back to the physical playfield, as seen in Figure 1. The

robot visualizes its moves by motion and sounds and reacts to the moves of its computer opponent by looking at the appropriate positions on the playfield.

Deliberately provoking positive and negative rewards from a user is only possible for the robot within a task where the human and the robot have the same understanding of which moves are desirable or undesirable. As the robot does not actually understand commands from its user at the beginning of the task, the user's commands as well as positive and negative feedback need to be reliably predictable from the task-state. In that case the robot can easily explore the user's reward behavior by performing in a good or bad way. Even though the combination of Hidden Markov Models and classical conditioning is designed to be robust against occasional false training examples it is desirable to keep their number as low as possible. In order to ensure that a good move of the robot will receive positive reward and a bad move will receive negative reward the games used for training must be designed in a way that the situation is easy to evaluate by the user. We assess the suitability of the different training tasks in the experiments described in section 6 of this paper.

4.1 Advantages of Virtual Training Tasks

Using virtual training tasks as a basis for human-robot-communication has different benefits. As mentioned at the beginning of this paper, one main advantage is the reduction of effort needed to implement perception and understanding of the environment, so that priority can be given to the system capabilities that are actually needed for interacting with a human.

Many commercially available robots used in research such as the AIBO or Khepera are quite small and have no or very simple actuators. So their ability to actually manipulate objects in their environment is often quite limited. AIBO, the robot used in our experiments, can only pick-up small cylindrical objects with its mouth and needs to approach them extremely precisely in order to be able to pick them up.

Another difficulty in real-world tasks is to detect errors during task-execution such as failing to pick up an object, hitting any objects that are in the way etc. Failing to detect that an attempted action could not be performed successfully poses a risk for misinterpreting the current status of the task and misunderstanding user interaction.

For these reasons, we decided to implement the training task in a way that the robot can complete it without having to directly manipulate its environment. When using a computer-based task, the current situation of the robot can be assessed instantly and correctly by the software at any time. It can be manipulated freely, e.g. to ensure exactly the same conditions for all participants in an experiment.

4.2 Selected Game Tasks

The following tasks were selected to be used in our experiments, because they are easy to understand and allow the user to evaluate every move instantly. We selected four different tasks in order to see whether different properties of the

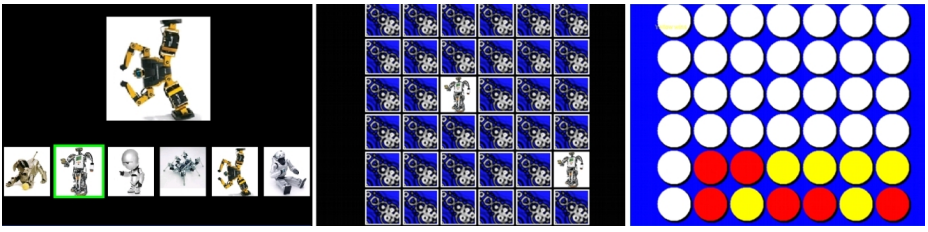

Fig. 2. Screenshots of the Virtual Game Tasks

task, such as the possibility to provide not only feedback but also instruction, the presence of an opponent or the game-based nature of the tasks influence the user's behavior. We implemented them in a way that they require little time-consuming walking movement from the robot. Screenshots of the playfields can be seen in figure 2.

Find Same Images. In the "Find Same Images"-Task, the robot had to be taught to chose the image, that corresponds to the one, shown in the center of the screen, from a row of six images. While playing, the image that the robot is currently looking or pointing at is marked with a green or red frame to make it easier for the user to understand the robot's viewing or pointing direction. By waving its tail and moving its head the robot indicates that it is waiting for feedback from its user. The participants were asked to provide instruction as well as reward to the robot to make it learn to perform the task correctly. The system was implemented in a way that the rate of correct choices and the speed of finding the correct image increased over time.

Pairs. In the "Pairs" game, the robot plays the game "Pairs": At the beginning of the game, all cards are displayed upside down on the playfield. The robot chooses two cards to turn around by looking and pointing at them. In case, they show the same image, the cards remain open on the playfield. Otherwise, they are turned upside down again. The goal of the game is to find all pairs of cards with same images in as little draws as possible. The participants were asked not to give instruction to the robot, which card to chose but teach the robot to play the game by giving positive and negative feedback only.

Connect Four. In the "Connect Four" game, the robot plays the game "Connect Four" against a computer player. Both players take turns to insert one stone into one of the rows in the playfield, which then drops to the lowest free space in that row. The goal of the game is, to align four stones of one's own color either vertically, horizontally or diagonally. The participants were asked to not to give instructions to the robot but provide feedback for good and bad draws in order to make the robot learn how to win against the computer player. Judging whether a move is good or bad is considerably more difficult in the "Connect Four" task than in the three other tasks as it requires understanding the strategy of the robot and the computer player.

Dog training. In the "Dog Training" task, the participants were asked to teach the speech commands "forward", "back", "left", "right", "sit down" and "stand up" to the robot. The "Dog Training" task is the only task that is not game-like and does not use the "virtual playfield". Only in this task the robot was remote-controlled to ensure correct performance. It was used by us as a control task in order to detect possible differences in user behavior between the virtual tasks and "normal" Human-Robot-Interaction.

5 The Learning Method

We propose a learning method consisting of two stages to allow the system to adapt to the user's way of giving positive as well as negative feedback. It combines an unsupervised low-level learning stage based on Hidden Markov Models (HMMs) with a supervised learning stage based on a mathematical model of classical conditioning. In the low-level "reward recognition learning" learning stage the system trains HMMs to match perceived utterances and prosodic patterns. In the high-level learning stage, the "reward association learning", the system creates associations between the trained models and either positive or negative rewards.

In this paper we are presenting results of the learning algorithm for understanding speech (utterances) and touch, combining the data from these two modalities for reliable recognition. Extensions are currently under development to deal with gesture as well as prosody of human speech. Different aspects were considered when choosing the combination of HMMs and classical conditioning for the purpose of learning to understand human feedback.

By combining unsupervised clustering of similar perceptions with a supervised learning method, such as classical conditioning, our system can learn the meaning of feedback from the user during natural interaction because the learning algorithm does not require any explicit information, such as transcriptions of the user's utterances or gestures. It only needs the information of whether an utterance means positive or negative feedback, which is determined by the training task.

HMMs usually show high performance for the classification of time series data and are therefore widely considered state-of-the-art for this purpose. Although HMMs are typically trained in a supervised way, different approaches for an unsupervised training of HMMs have been described in literature [7].

We chose conditioning as a biologically inspired approach which typically converges quickly and has other desirable properties, which are described in section 5.2. Classical conditioning allows the system to weigh and combine user inputs in different modalities according to the strength of their association toward positive or negative reward. An overview of the learning algorithm that is used to train the HMMs and associations is shown in Figure 3. It is described in detail in sections 5.1 and 5.2.

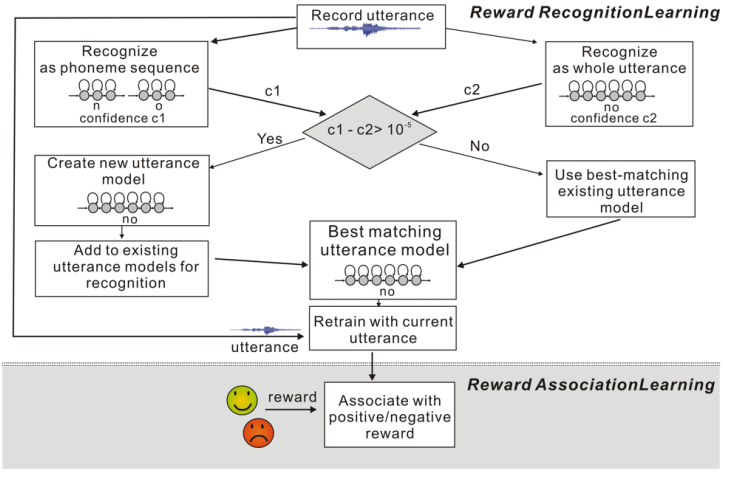

Fig. 3. Flowchart of the Algorithm

5.1 Reward Recognition Learning

The basis of the reward recognition learning are sets of pre-trained elementary Hidden Markov Models (HMMs) as well as a model of possible touch patterns. HMMs are employed for the low-level modeling of perceptions. As a standard approach for the classification of time series data, HMMs are widely used in literature. The use of Mel-Frequency-Cepstrum-Coefficients (MFCC) for HMM-based speech recognition is described in [12]. They are used in our work as an input for the HMM-based low-level learning phase.

The initial HMM-set for learning speech-based rewards contains all Japanese monophones and is taken from the Julius Speech Recognition project [13]. We use standard left-right HMMs for recognition. The models base on MFCC feature-vectors generated from the recorded speech data. We decided to use monophone models instead of diphone or triphone models although the latter are more powerful and widely used in speech recognition, because of their smaller number and lower complexity. While the monophone set for Japanese contains 43 models, 7946 HMMs are contained in the Julius triphone set for Japanese. As the initial HMMs only form a basis for constructing word models and training them in a user-dependent way, perfect accuracy is not needed in this stage. Moreover, the number of states of our word models directly depends on the number of states of the concatenated elementary models, which is significantly higher for triphone models. To keep the number of necessary training utterances low, the degrees of freedom, that is the number of states and transitions, used when training the models should not grow excessively large.

We use a grammar for the phoneme recognizer that permits an arbitrary sequence of phonemes, not restricted by a language dependent dictionary. A sequence of phonemes may have an optional beginning and ending silence and

contain short pauses. The grammar of our utterance model allows exactly one utterance with an optional beginning or ending silence.

During the training phase, utterances from the user are detected by a voice activity detection based on energy and periodicity of the perceived audio signal.

Every time a reward from the user is observed, first the system tries to recognize the utterance with the phoneme sequence recognizer as well as with the recognizer for the already trained utterance models. Matching is done by HVite, an implementation of the Viterbi Algorithm included in the Hidden Markov Model Toolkit (HTK) [12]. The result of this first step of the reward recognition learning is the best-matching phoneme sequence and the best matching utterance out of the utterance models that have been generated up to that point. In addition to that, a confidence level is output by the system for both recognition results. The confidence level, that is, the log likelihood per frame of both results calculated by HVite, is compared to find out whether to generate a new model or retrain an existing one. If the confidence level of one of the existing models matches the utterance well enough, that is, the confidence level of the best-fitting phoneme sequence is less than 10^{-5} better than the confidence level of the best-fitting existing utterance model then the best-fitting utterance model is retrained with the new utterance.

If the confidence level of the best-matching phoneme sequence is more than 10^{-5} better than the one of the best-fitting whole-utterance model, then a new utterance model is initialized for the utterance. The new model is created by concatenating the HMMs that make up the recognized most likely phoneme sequence. The new model is retrained with the just observed utterance and added to the HMM-set of the whole-utterance recognizer. So it can be reused when a similar utterance is observed. The threshold of 10^{-5} was determined experimentally, using data that was recorded with the same audio equipment but not used for training or evaluation.

As for touch-based rewards, we decided after the experiments to abandon using complex and time-consuming HMM based modeling for the time being and decided to model touch by the following three patterns for touching the head sensor and touching the back sensor.

- Touching the robot's sensor one or multiple times for less than half a second (hitting)
- Touching the robot's sensor for more than a second one or multiple times (stroking)
- Touch-based interaction not falling into one of the above classes

The HMM or touch-pattern that this low-level classification and learning stage outputs is the current most accurate available model of the observed reward. It serves as an input for the reward association learning where it is associated with either positive or negative meaning.

5.2 Reward Association Learning

In the reward association learning an association between the HMM obtained from the reward recognition learning and either positive or negative feedback is

created or reinforced. The information of whether the HMM should be associated with positive or negative reward is obtained from the current state of the game. If the last move of the robot was a good one, the observation is associated with positive reward. If the last move was a bad one, the observation is associated with negative reward.

Reward association learning is based on the theory of classical conditioning, which was first described by I. Pavlov and originates from behavioral research in animals. In classical conditioning, an association between a new, motivationally neutral stimulus, the so-called conditioned stimulus (CS), and a motivationally meaningful stimulus, the so-called unconditioned stimulus (US), is learned [4].

Classical conditioning possesses several relevant features, such as blocking, extinction, sensory preconditioning and second-order conditioning, that allow our system to give priority to rewards that are used most frequently, adapt to changes in reward behavior and associate rewards which often occur together. These properties are explained in more detail in [3].

The Rescorla-Wagner-Model of Classical Conditioning. There are several mathematical theories, trying to model classical conditioning as well as the various effects that can be observed when training real animals using the conditioning principle. The models describe how the association between an unconditioned stimulus and a conditioned stimulus is affected by the occurrence and co-occurrence of the stimuli. In this study, the Rescorla-Wagner model [4], which was developed in 1972 and has served as a foundation for most of the more sophisticated newer theories is employed. In the Rescorla-Wagner model, the change of associative strength of the conditioned stimulus A to the unconditioned stimulus US(n) in trial n, $\Delta VA(n)$, is calculated as in (1).

$$\Delta VA(n) = \alpha A \beta US(n)(\lambda US(n) - Vall(n)) \tag{1}$$

αA and $\beta US(n)$ are the learning rates dependent on the conditioned stimulus A and the unconditioned stimulus $US(n)$ respectively, $\lambda US(n)$ is the maximum possible associative strength of the currently processed CS to the nth US. It is a positive value if the CS is present when the US occurs, so that the association between US and CS can be learned. It is zero if the US occurs without the CS. In that case, $\Delta VA(n)$ becomes negative. Thus, the associative strength between the US and the CS decreases. $Vall(n)$ is the combined associative strength of all conditioned stimuli toward the currently processed unconditioned stimulus. The equation is updated on each occurrence of the unconditioned stimulus for all conditioned stimuli that are associated with it.

One advantage of using conditioning as an algorithm for learning the associations between positive/negative reward and the user's corresponding behaviors is its rather quick convergence, depending on the learning rate.

In this study, the learning rates for conditioned and unconditioned stimuli are fixed values for each modality but can be optimized freely. They determine how quickly the algorithm converges and how quickly the robot adapts to a change in reward behavior. The maximum associative strength is set to one, in case the

corresponding CS is present, when the US occurs, zero otherwise. The combined associative strength of all conditioned stimuli toward the unconditioned stimulus can be calculated easily by summarizing the association values of all the CS toward the US, that have been calculated in the previous runs of the reward recognition learning.

6 Experiments

We experimentally evaluated our training method as well as our learning algorithm. Ten persons participated in our study. All of them were Japanese graduate students or employees at the National Institute of Informatics in Tokyo. Five of them were females, five males. The age of the participants ranged from 23 to 47. All participants have experience in using computers. Two of them have interacted with entertainment robots before. Interaction with the robot was done in Japanese. During the experiment, we recorded roughly 5.5 hours of audio and video data containing 533 rewards which consisted of 2409 individual stimuli. Figure 4 shows a scene from the video taken during the experiments.

6.1 Results

We evaluated the performance of the learning algorithm offline with the data recorded within the above described experimental setting. The system was trained and evaluated with data from the "Find Same Images" and the "Pairs" task. The data from the "Connect Four" task was not used because the participants often were not able to evaluate whether a move was good or bad. Therefore reward from the user was observed for less than one third of the robot's moves in the "Connect Four" task, had a strong positive bias and often did not match the judgment from the evaluation function of the game. We also excluded the data

Fig. 4. Participant instructing AIBO

Table 1. Confusion Matrix (in percent)

	Positive(actual)	Negative(actual)
Positive (recognized)	48.32	4.49
Negative (recognized)	5.18	42.01

from the "Dog Training" task where the robot was remote-controlled. Training and evaluation were done in a user-dependent way using leave-one-out cross evaluation in order to use as much data for training and evaluation as possible. The average accuracy of our system for classifying between positive and negative rewards given by one user based on speech and touch was 90.33%. The standard deviation between users was 3.41%. As the rewards given by the participants showed a slight bias toward positive feedback, the confusion matrix, shown in Table 1 gives a more detailed overview over the performance of our recognizer. Using speech only we reached a recognition rate of 78.35% with a standard deviation of 4.37%. Using touch only the recognition rate was 76.16% with a high standard deviation of 16.92% as the usage and frequency of touch varied strongly between users. Typically one reward consists of multiple stimuli. A stimulus is one utterance or one touch of the touch sensors. The recognition rate for individual uncombined speech and touch stimuli is 80.20% with a standard deviation of 3.46%. This is about 10% lower than the recognition rate for combined rewards shown above. These results underline that combining stimuli given through different modalities is crucial for a reliable recognition.

A more detailed analysis on the participants' behavior during the interaction with the robot in the four training tasks is presented in [2]. We found, that the most frequently used modality was speech, which accounts for 78.37% of the recorded stimuli, followed by touch, which accounts for 20.92% of the stimuli. Gesture was almost not used (0.71%) for giving reward, although it was frequently used for providing instruction to the robot. The preferred utterances to give positive and negative feedback varied among different people as well as for one person but we did not observe a strong task-dependence.

We prepared a questionnaire for the participants to ask about their evaluation of the different tasks. They could rate their agreement with different statements concerning the interaction on a scale from one to five, where one meant "completely agree" while five meant "completely disagree". The results can be found in table 2. As can be seen from the table, the four tasks were considered almost equally enjoyable by the participants. For the "Find same Images" task and the "Dog Training" task, the participants' impression that the robot actually learned through their feedback and adapted to their way of teaching was strongest. Those two tasks allowed the participants to not only give feedback to the robot but also provide instructions. Moreover, they were designed in a way that the robot's performance improved over time. In the "Dog Training" task, the robot was remote-controlled to react to the user's commands and feedback in a typical Wizard of OZ-Scenario. However, in the "Find Same Images" task, which was judged almost equally positively by the participants, the user's

Table 2. Results of the Questionnaire (standard deviation given in brackets)

	Same	Pairs	Four	Dog
Teaching the robot through the given task was enjoyable	1.81 (1.04)	1.90 (0.83)	1.81 (0.89)	1.63 (0.81)
The robot understood my feedback	1.27 (0.4)	1.81 (0.74)	2.90 (0.85)	1.81 (0.30)
The robot learned through my feedback	1.36 (0.59)	2.81 (0.93)	3.45 (0.95)	1.54 (0.69)
The robot adapted to my way of teaching	1.45 (0.66)	2.63 (1.05)	3.45 (1.04)	1.64 (0.58)
I was able to teach the robot in a natural way	2.18 (0.96)	2.09 (0.86)	2.54 (1.12)	1.64 (0.69)
I always knew, which instruction or reward to give to the robot	2.00 (0.72)	2.09 (0.86)	2.90 (1.02)	1.91 (0.83)

instructions and feedback were not actually understood by the robot but anticipated from the state of the training task. This did not have a negative impact on the participants impression that the robot understood their feedback, learned through it and adapted to their way of teaching. The lowest ratings were given for the "Connect Four" task. As the robot's moves could not be evaluated as easily, as in the other tasks, the participants were unsure which rewards to give and therefore did not experience an effective teaching situation. This also becomes apparent in the overall low quantity of feedback given in this task which still included incorrect feedback.

7 Conclusion

In this paper, we described and evaluated a method for learning a user's feedback for human-robot-interaction. The performance based on interpreting speech and touch rewards from a human can be considered sufficiently reliable for being used to teach a robot by reinforcement learning.

Training tasks for learning to understand rewards need to be carefully designed to ensure that the robot's moves can be easily evaluated by the user. In a strategic game like "Connect Four" it is difficult to instantly assess whether a move was good or bad. This results in a decrease of the quantity as well as the correctness of the rewards and also affects the user experience.

The reliability of recognizing reward could be enhanced by not only processing the speech utterances but also taking into account prosody. For learning to interpret commands, other than rewards, gesture recognition will be helpful, so integrating prosody and gesture as additional modalities into our system is the current priority of our ongoing research.

One important question that remains open after the study is the similarity of user behavior between virtual tasks and real world tasks. Although differences in giving positive and negative reward between the virtual game tasks and the

dog training task could not be observed this does not necessarily mean that it is generally possible to train a robot for a real world task using a virtual task. This question will be targeted in a follow-up study.

References

1. AIBO Remote Framework, http://openr.AIBO.com
2. Austermann, A., Yamada, S.: Good Robot, Bad Robot - Analzying User's Feedback in a Human-Robot Teaching Task. In: Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication 2007 (RO-MAN 2008) (2008)
3. Austermann, A., Yamada, S.: Learning to Understand Multimodal Rewards for Human-Robot-Interaction using Hidden Markov Models and Classical Conditioning. In: Proceedings of the IEEE World Congress of Computational Intelligence (WCCI 2008) (2008)
4. Balkenius, C., Morn, J.: Computational models of classical conditioning: a comparative study. In: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior (1998)
5. Ballard, D.H., Yu, C.: A multimodal learning interface for word acquisition. In: 2003 Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (2003)
6. Iwahashi, N.: Active and Unsupervised Learning for Spoken Word Acquisition Through a Multimodal Interface. In: RO-MAN 2004 13th IEEE international workshop on robot and human interactive communication (2004)
7. Li, C., Biswas, G.: A Bayesian Approach to Temporal Data Clustering using Hidden Markov Models. In: Proceedings of the Seventeenth International Conference on Machine Learning 2000, pp. 543–550 (2000)
8. Kayikci, Z.K., Markert, H., Palm, G.: Neural Associative Memories and Hidden Markov Models for Speech Recognition. In: IJCNN 2007 Conference Proceedings (2007)
9. Nogueiras, A., Moreno, A., Bonafonte, A., Marino, J.B.: Speech Emotion Recognition Using Hidden Markov Models. In: Proceedings of Eurospeech (2001)
10. Steels, L., Kaplan, F.: AIBO's first words: The social learning of language and meaning. Evolution of Communication 4(1) (2001)
11. Xin, M., Sharlin, E.: Sheep and wolves: test bed for human-robot interaction. In: CHI 2006 Extended Abstracts on Human Factors in Computing Systems, Montreal, Quebec, Canada, April 22 - 27 (2006)
12. Young, S., et al.: "The HTK Book" HTK Version 3 (2006), http://htk.eng.cam.ac.uk/
13. The Julius Speech Recognition Project, http://julius.sourceforge.jp/