# A Biologically Inspired Approach to Learning Multimodal Commands and Feedback for Human-Robot Interaction

**Anja Austermann**
The Graduate University for
Advanced Studies (SOKENDAI)
2-1-2 Hitotsubashi, Chiyoda-ku
101-8430, Tokyo, JAPAN
anja@nii.ac.jp

**Seiji Yamada**
National Institute of Informatics
The Graduate University for
Advanced Studies (SOKENDAI)
2-1-2 Hitotsubashi, Chiyoda-ku
101-8430, Tokyo, JAPAN
seiji@nii.ac.jp

*CHI 2009*, April 4 – 9, 2009, Boston, Massachusetts, USA.
ACM 978-1-60558-247-4/09/04.

## Abstract

In this paper we describe a method to enable a robot to learn how a user gives commands and feedback to it by speech, prosody and touch. We propose a biologically inspired approach based on human associative learning. In the first stage, which corresponds to the stimulus encoding in natural learning, we use unsupervised training of HMMs to model the incoming stimuli. In the second stage, the associative learning, these models are associated with a meaning using an implementation of classical conditioning. Top-down processing is applied to take into account the context as a bias for the stimulus encoding. In an experimental study we evaluated the learning of user feedback with our learning method using special training tasks, which allow the robot to explore and provoke situated feedback from the user. In this first study, the robot learned to discriminate between positive and negative feedback with an average accuracy of 95.97%.

## Keywords

Human-Robot-Interaction, Speech Perception, Machine Learning, User Feedback, Multimodality

## ACM Classification Keywords

H.5.2 [Information Interfaces and Presentation]: User Interfaces

## Introduction

A major challenge for creating robots that are easily accessible for everyone is to enable them to understand natural user interaction.  Different approaches to this problem can be found in literature [3][6]. We are taking a step towards this goal by enabling a robot to learn a user's preferred way of giving commands and reward through training. A lot of research has been done on automatic symbol grounding for a robot. We chose a different approach and assume that the robot already has a symbolic representation of the actions, that it is able to perform, and the objects or places, it can recognize and that it knows how to make use of positive and negative feedback. This is likely to be the case for typical service- or entertainment robots. Our goal is to associate these existing symbolic representations with commands, object names or feedback given naturally using speech, prosody and touch, to enable the robot to deal with instruction given by the user in his or her preferred way. To reach this goal, we use a combination of special training tasks, which allow the robot to provoke commands and feedback from the user, and a two-staged learning algorithm, which has been designed to resemble the processes, which occur in human associative learning.

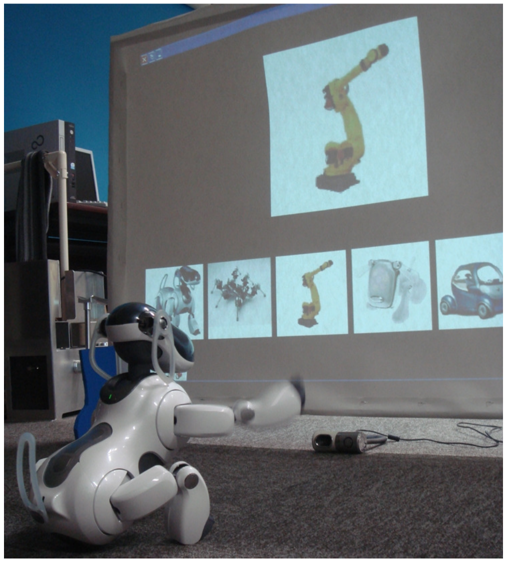

Fig.1: Aibo Performing Training Task

## Related Work

There has been a great deal of research on learning through human feedback in recent years. One example that is particularly related to our work is presented in [6]. Kim and Scassellati describe an approach to recognize approval and disapproval in a Human-Robot teaching scenario and use it as an input for Q-Learning. They use a single-modal approach based on prosody. Iwahashi describes an approach [3] to the acquisition of new words for the multimodal interface of a robot. He applies Hidden Markov Models to learn verbal representations of objects and motions perceived by a stereo camera. The robot interacts with its user to avoid and resolve misunderstandings.

## Training Tasks

The robot learns to understand the user's commands and feedback in a training phase. The design of the training tasks is a key point for our learning method because they enable the robot to provoke commands and feedback from the user.  We decided to use computer-based "virtual" training tasks to avoid time-consuming walking movement of the robot and to enable the robot to access all properties of the task instantly. We implemented a framework, which can easily be extended to fit different tasks, robots or virtual characters. The task-scene is projected to a white screen and the robot visualizes its actions by motion, sounds and its LEDs. Fig. 1 shows the robot performing a "virtual" task. During the training the robot cannot actually understand its user but needs to react appropriately to ensure natural interaction. This is done by designing the training task in a way that the robot can anticipate the user's commands and feedback. In our experiment for learning feedback, which is described at the end of this paper, we used game-based tasks where both the robot and the user could determine easily, whether a move was good or bad. This allowed the robot to provoke positive or negative feedback by making good or bad moves. For learning object names and commands, we use an animated "virtual" living room. The robot can query the user for object names by pointing at them. Appropriate animations are shown on the screen to facilitate understanding for the user. After learning the names of objects and places, the robot continues with learning command patterns like "switch on <object>", "move <object> to <place>" etc. In order to enable the robot to learn, the system needs to make the user give commands in his preferred way but with a predefined meaning. This is done by showing situations in the virtual living room, where it is obvious which task needs to be performed by the robot. (e.g. it is getting dark and the light is still switched off)

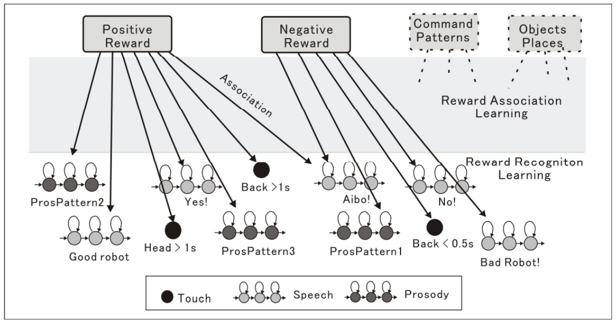

Fig. 2: Structure Created by the Learning Algorithm

## Learning Method

We implemented a learning method, which is inspired by human associative learning and speech perception. [2] Like in natural learning by conditioning, the learning process is divided into a stimulus encoding phase and an associative learning phase. Our implementation of the stimulus encoding is based on unsupervised training of Hidden Markov Models (HMMs) to cluster stimuli that are likely to belong to the same utterance or a similar prosody pattern. The associative learning phase uses classical conditioning. It establishes associations between the HMMs for the encoded stimuli and meanings, such as positive/negative feedback or names of objects. Perception is no unidirectional process. In addition to the stimulus-driven bottom-up processes, top-down processes integrate context information to find the best possible interpretation of a stimulus. We implemented top-down processes by using the learned associations and the knowledge about the state of the training task. The associative strength between an HMM and the expected command, object/place descriptor or feedback is used as a bias when determining the best HMM for retraining. For example HMMs, which already have an existing association to positive feedback become more likely to be selected when positive feedback is expected again. A sample structure, created by the learning algorithm, is shown in Fig. 2.

### *Encoding of Speech Stimuli*

The stimulus encoding for speech stimuli creates and retrains user-dependent utterance models starting from an existing set of monophone HMMs, which is taken from the Julius speech recognition project [7]. As the robot learns automatically through interaction, no transcription of the utterances is available. Therefore, an unsupervised clustering of perceived feedbacks that are likely to correspond to the same utterance is necessary. This is done using two recognizers in parallel. One recognizer tries to recognize the observed utterance as a phoneme-sequence. The other recognizer uses the already trained utterance models to determine the best-matching known utterance. Every time a feedback from the user is observed, first the system tries to recognize the utterance with both recognizers. The recognizers return the best-matching phoneme sequence and the best matching utterance along with confidence levels. The confidence levels are compared to decide whether to generate a new model or retrain an existing one. Typically, for an unknown utterance, the phoneme-sequence based recognizer returns a result with a noticeably higher confidence, than the one of the best matching utterance model. In this case a new utterance model is created by concatenating the HMMs of the recognized most likely phoneme sequence. The new model is retrained with the utterance and added to the HMM-set of the utterance recognizer to be reused when a similar utterance is observed. For a known utterance, the confidence level of the corresponding utterance model is either higher or close to the one of the best-matching phoneme-sequence. In that case the overall best-fitting utterance model is determined, taking into account the bias from the top-down processing. The model is then retrained with the new utterance.

We distinguish three different kinds of utterances, that the speech stimuli encoding needs to deal with: positive/negative feedback, names of objects/places and command-patterns. Command-patterns can have a variable number of slots for inserting object- or place-names like “Stand up”, “Get <object>”or “Can you move <object> to <place>?”. An example command structure is shown in Fig. 3. The leaves of the tree are trained HMMs. The inner nodes are symbolic representations of objects and command patterns. The thick lines are associations, learned in the associative learning phase. Feedback-utterances, names and command-patterns without any slots can be trained as single HMMs. In case of command patterns with one or more slots, the system first needs to determine which parts of the utterance belong to the verb pattern and which parts belong to object/place names: The system knows the meaning of the command that the user is going to utter and which objects are involved from the training task. The algorithm uses this information to locate object/place names in the utterance by matching the utterance against all HMMs that have an existing association to the expected objects. This is why the training needs to start from learning object/place names before learning the command-patterns. The utterance is then cut at the boundaries of the detected names. All parts that do not belong to the name of an object or place are expected to belong to the

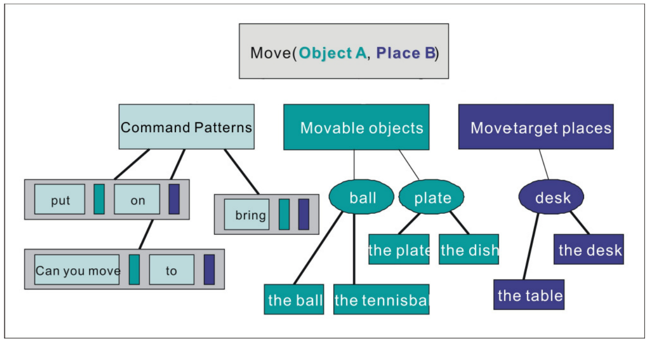

Fig. 3: Structure of a Command

command-pattern and used to create or retrain HMMs as described above. The places, where object- or place-names have been cut out are modeled as slots in the grammar of the utterance recognizer

*Encoding of Prosody-Based Stimuli*

The HMMs for interpreting prosody are based on features [4] extracted from the speech signal. First, the signal is divided into frames. For each frame, a feature vector is calculated, containing the pitch, the pitch difference to the previous frame, the energy, the energy difference to the previous frame and the energy in frequency bands 1-n. The sequence of feature vectors is used for training the HMMs. Additionally, the algorithm calculates some global information based on all frames belonging to one utterance. These are the average, minimum and maximum, range and standard deviation and the average difference between two frames for pitch as well as energy. For determining, which HMM is trained with which utterances, the system relies on these global features. Utterances with similar global features are clustered and one HMM is trained for each cluster. This model is then passed to the associative learning.

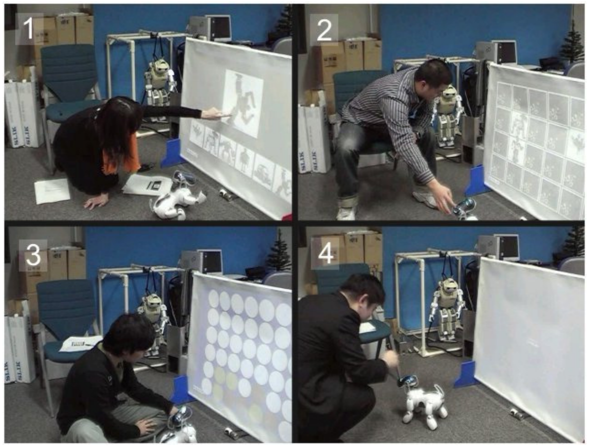

Fig. 4: Scenes from the Experiment

*Encoding of Touch-Stimuli*

We decided to use a simple duration based model for encoding touch. A touch of the head or back sensor of the robot can fall into one of three categories:

- short: less than 0.5 seconds
- medium: between 0.5 seconds and 1 second
- long: one second or longer

Typically, short touches were observed when the user was hitting the robot, while medium and long touches corresponded to caressing or stroking the robot.

*Associative Learning*

We use the Rescorla-Wagner model [5] of classical conditioning to associate HMMs with the existing symbolic representations of commands, objects or feedbacks. The symbolic representations are used as unconditional stimuli. The HMMs, encoding stimuli from the user, are used as conditioned stimuli. Classical conditioning has different desired properties, such as blocking, secondary conditioning and sensory preconditioning which allow the system to integrate and weight stimuli from different modalities, emphasize salient stimuli and establish connections between multimodal conditioned stimuli, e.g. between certain utterances and prosody patterns.

## Experimental Evaluation

We performed a first experimental evaluation of our training method and learning algorithm in a user study with 10 participants (5 male / 5 female). All of them were students or employees at the National Institute of

Informatics in Tokyo. The experiment focused on learning to understand *positive and negative feedback*. We used four different game-like training tasks, which are shown in Fig.4 and described in more detail in a user study in [1]. In the first task, the robot had to select the image, which corresponded to a sample. In the second task the robot played the game Pairs. In the third task, the robot played the game Connect Four against a computer player. In the fourth task, the robot learned five different fixed commands such as “stand up”, “sit down”, etc. from the user. The tasks varied in difficulty and in how freely the user was allowed to interact with the robot. The participants were asked to give instruction and feedback to the robot freely in their preferred way and they were told that the robot learns through their feedback. The experimental setting is shown in Fig. 5. Evaluation was done using 10-fold cross evaluation. After training, the algorithm reached an average accuracy of 95.97% (sd=3.30%) for discriminating positive and negative feedback based on multimodal integration of speech, prosody and touch. This is considerably higher than the individual recognition accuracies of 83.53% (sd= 8.30%) for speech, 84.27% (sd=8.57%) for prosody and 88.17% (sd=11.77%) for touch.

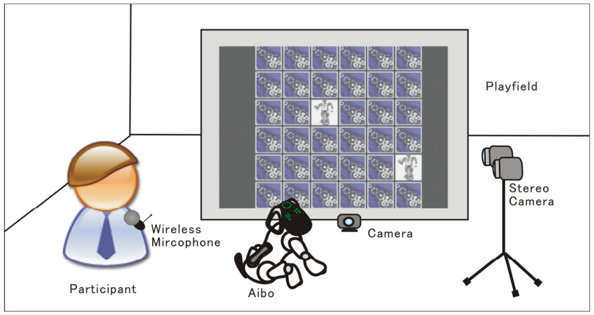

Fig. 5: Experimental Setting

## Conclusion and Further Work

We proposed a concept for learning natural commands and feedback from a user through a training task and showed first results for understanding positive and negative reward. While understanding feedback already enables the robot to learn from the user, for instance through reinforcement learning, it is necessary for a more convenient use that the robot can also learn more general commands. Our next step will be to conduct follow-up experiments for using our algorithm to learn typical commands. Currently, the system only recognizes speech, touch and prosody. As pointing gestures are frequently used in human communication, integration of pointing gestures and reference words will be a focus of our future research.

## References

[1] A. Austermann, S. Yamada: ““Good Robot, Bad Robot” - Analzying User’s Feedback in a Human-Robot Teaching Task”, In Proc. of the RO-MAN 2008, 41-46

[2] D. Groome: An Introduction to Cognitive Psychology. Psychology Press, Second Edition, 2008

[3] N. Iwahashi: "Robots that learn language - Developmental Approach to Human-Machine Conversations" Proc. EELC 2006, 142-179, 2006.

[4] T. L. Nwe, S. Foo, S. Wei; L. De Silva, "Speech emotion recognition. using hidden Markov models", Speech communication 41,4, 2003

[5] R. Rescorla, A. Wagner: “A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement.”, Classical Conditioning II, Appleton Century Crofts, 64-99, 1972

[6] Kim, B. Scassellati, "Learning to Refine Behavior Using Prosodic Feedback", In Proc. of the ICDL 2007, pp. 205-210

[7] The Julius Speech Recognition Project: http://julius.sourceforge.jp