Taming robots with clicker training : A solution for teaching complex behaviors

Frédéric Kaplan (1), Pierre-Yves Oudeyer (1), Enikö Kubinyi (2) and Ádám Miklósi (2)

(1) Sony CSL Paris, 6 rue Amyot 75005 Paris, France.
(2) Department of Ethology, Eötvös University, Göd, Hungary

kaplan@csl.sony.fr, oudeyer@csl.sony.fr, kubinyie@freemail.hu, miklosa@ludens.elte.hu

Abstract. In this paper we want to propose the idea that some techniques used for animal training might be helpful for solving human robot interaction problems in the context of entertainment robotics. We present a model for teaching complex actions to an animal-like autonomous robot based on "clicker training", a method used efficiently by professional trainers for animals of different species. After describing our implementation of clicker training on an enhanced version of AIBO, Sony's four-legged robot, we argue that this new method can be a promising technique for teaching unusual behavior and sequences of actions to a pet robot.

1 Introduction

The recent years have been characterised by the expansion of animal-like entertainment robots (Kusahara, 2000; Druin and Hendler, 2000; Kaplan, 2001). The AIBO, commercialized by Sony in 1999, was the first product of this new generation of robots (Fujita and Kitano, 1998). Its main originality is to be both an autonomous robot and a digital creature. As an autonomous robot, the AIBO is designed to move and behave in unknown environments. But as a digital creature, it is not meant to perfom service tasks for its owner. It will not do something "useful" and for this very reason, it may actually be a companion with whom it is pleasant to interact (Kaplan, 2000a).

Interactions with current entertainment robots is still very restrained. These robots act autonomously without paying much attention to their owner. One of the challenges and pleasures in keeping a real pet, like a dog, is that the owner has to "tame" it. A dog owner is proud when he has the impression that his pet changes its own behavior according to his teaching. We believe this is also a way for an interesting relationship to emerge between an entertainment robot and its owner (Dautenhahn develops a similar argument in (Dautenhahn, 1999))). For this reason, a growing number of research groups are currently focusing on teaching techniques for autonomous robots (Billard et al., 1998; Roy, 1999; Kaplan, 2000b; Fujita et al., 2001).

This paper focuses on a method for teaching actions to an animal-like entertainment robot. Of course, the simplest way would be to allow the owner to program directly new actions for the robot. But for the purpose of entertainment robotics it would be much more interesting if this teaching would take place only through interactions, as it does with real pets.

We believe that, for this matter, a collaboration between robot builders and ethologists can be interesting. Exchanges between ethologists and robotic engineers have several times proven to be fruitful in the past (see (Steels and Brooks, 1994), (Arkin, 1998) and (Webb, 2000) for instance). But apart from the exception of Blumberg's team at MIT Media Lab (Blumberg et al., 1996; Blumberg, 1997; Yoon et al., 2000), the field of animal training has not yet been much investigated as a source of inspiration of robotics researchers.

How to teach complex behaviors (and commands associated to them) only through interactions is a particularly hard problem to tackle. We argue that, in the context of entertainment robotics, the difficulties encountered when trying to teach an autonomous pet robot a complex behavior are similar to the ones met by animal trainers. When a trainer wants to teach a dolphin to do a special jump on command, he cannot show it or explain it what to do. The animal needs to discover the action by itself. If it is a behavior that the animal performs often, it will not be too difficult. The trainer will simply need to indicate : "This is it, this is what I wanted." But if it is a rare and complex behavior, the trainer will need to guide the animal. The constraints are very similar in our context. The robot needs to discover by itself what its owner wants. Therefore, some techniques used for pet training might be helpful for solving this problem in robotics.

Among all the training techniques currently used, the clicker training method has proven to be one of the most efficient for a large variety of animals including dogs, dolphins and chickens (Pryor, 1999; Tillman, 2000). In this paper, we intend to show that it can also be used for the training of autonomous robots.

The next section presents different methods for teaching actions in the context of animal training and robotics. We discuss why clicker training seems to be a promising way of handling the problem of rare behaviors and sequences of actions. Then, we explain briefly the principles of clicker training. The following session describes our implementation of a first prototype of a training session with AIBO, Sony's four-legged robot. The last session discusses related work, experimental observations, limitations and possible extensions of the model.

2 Methods for teaching actions

We will start this quick review of methods for teaching actions to both animals and robots, by mentioning an error commonly observed during amateur training sessions. Many a dog owner makes the mistake to chant commands while attempting to put the dog in the desired position. For instance, the trainer repeats the word "SIT" while pushing the dog's rear down to the ground. The method fails to give good results for several reasons:

- The animal is forced to choose between paying attention to the trainer's word and learning a new behavior.
- As the command is repeated several times, the animal does not know to which part of its behavior the command is related.
- Very often the command is said before the behavior : for instance "SIT" is given while the animal is still in a standing position, so it cannot be associated with the desired sitting position.

For these reasons, most trainers decide to teach commands and behaviors separately. In practice, they teach the behavior first and then, add the command. Given that designing robots good at sharing attention and discriminating stimuli is very difficult, it is advisable to operate the same way when teaching an entertainment robot.

Consequently, our main problem is to obtain the production of the right behavior. We are now going to discuss briefly the performance of different techniques commonly used for teaching actions (see synthesis in Table 1).

The *modelling* method (the term "moulding" is also used), often tried by dog owners, is almost never used by professional trainers. It involves physically manupulating the animal into the desired position and giving positive feedback when it manages to reach it. With this method, the animal remains passive which might explain why most of the times learning performances are poor. Modelling has been mainly used to teach positions to robots in industrial contexts. As soon as the robot is autonomous and constantly active, its manipulation becomes problematic. Only partial modelling can be envisioned. For instance, the robot can sense that the trainer is pushing on its back and can decide to sit if programmed to do so. But it is not easy to generalize this method to complex movements involving more than just reaching a static position.

Luring (also called "magnet method") is similar to modelling except that it does not involve a physical contact with the animal. A toy or a treat is put in front of the dog's nose and the trainer can use it to guide the animal into the desired position. This method gives satisfactory results with real dogs but can only be used for teaching positions or very simple movement. Luring has not been used much in robotics. Commercial AIBOs are programmed to be automalically interested in red objects. In consequence, some robot owners use this tendency to guide their artificial pets into desired places. But this usage remains rather limited and no learning is involved. (Billard, 1998) reports experiments where robots learn about the interpretation of their sensorimotor perceptions by following a teacher robot or a human. This may be seen as a kind of elaborate luring.

In contrast with modelling and luring, *capturing* methods exploit behaviors that the animal performs spontaneously. For instance, every time a dog owner acknowlegdes his pet is in the desired position or performing the right behavior he or she can give a positive reinforcement. This indicates to the animal that it has just performed an interesting behavior. Several robotic systems in which a human selects behaviors in an autonomous robot's behavior repertoire have already been described (see for instance (Dautenhahn, 1999)). We also made a simple prototype to teach action to an autonomous robot using a capturing technique. We programmed the robot to perform autonomously successive random behaviors. Each time it was performing a behavior that we wanted to associate with a signal (for instance a word) we emitted the signal immediately afterwards. To teach the robot a word like 'sit', the trainer needs to wait for the robot to sit spontaneously. The main problem with this method is that it does not work when the number of behaviors that can receive a name is too large : the time needed to have the robot perform the right behavior by chance is too long.

Methods based on *imitation* are used very seldomly by animal trainers. One reason is that the animal anatomy is, in most cases, very different from ours. Very few animals appear to be able to imitate. This has been acknowledged only with "higher animals" (mostly primates, cetaceans and humans). However, in robotics, several research groups have been tackling the problem of imitation for the last five years (see for instance (Kuniyoshi et al., 1994; Dautenhahn, 1995; Bakker and Kuniyoshi, 1996; Billard et al., 1998)). In principle, imitation can handle the learning of sequences of actions and of very rare behaviors. Despite very interesting progress, imitation still needs good artificial vision techniques or special sensors to capture the movements to imitate. It is therefore difficult to envision it on currently available autonomous robots.

As imitation is too complex for most animals, animal trainers prefer an alternative technique called *shaping*. To shape a behavior, the trainer breaks it down in small achievable responses that will eventually lead to the final desired behavior. The main idea is to guide progressively the animal towards the right behavior. To perform each step, it is possible to use any of the techniques presented in this section. Several techniques can be used for shaping, but the most popular method is called clicker training.

3 A brief introduction to clicker training

Clicker training is based on B.F. Skinner's theory of operant conditioning (Skinner, 1938). In the 1980s Gary Wilkes, a behaviorist, collaborated with Karen Pryor, a dolphin trainer, to popularize this method for dog training (Pryor, 1999). The whistle traditionally used for dolphins was replaced by a little metal cricket (the clicker).

When you press the clicker, it emits a brief and sharp sound. This sound does not mean anything by itself for the animal. But the trainer can associate it with a *primary reinforcer*. Primary reinforcers are things that the animal instinctively finds rewarding such as food, toys, etc (it is sometimes referred as an unconditioned stimulus in animal learning literature). After having been associated a number of times with the primary reinforcer, the clicker will become a *secondary reinforcer* (also called *conditionned reinforcer*). Then it will act as a positive cue, meaning that a reward will come soon. Because the clicker is not the reward in itself, it can be used to guide the animal in the right direction. It is also a more precise way to signal which particular behavior needs to be reinforced. The trainer gives the primary reinforcer only when the animal performs the desired behavior. This signals the end of the guiding process.

The clicker training process involves at least four steps:

- Charging up the clicker : During this first process, the animal has to associate the click with the reward (the treat). This is achieved by clicking and then treating the animal consistently for around 20-50 times until it gets visibly excited by the sound of the clicker.

- Getting the behavior : Then the animal is guided to perform the desired action. For instance, if the trainer wants the dog to spin in a clockwise circle, it will start by clicking to the slightest head movement to the right. When the dog performs the head movement consistently, the trainer clicks only when it starts to turn its body to the right. The criteria is raised

Training techniques	Sequences of actions	Unusual actions	Usability with animals	Usability for autonomous robots
Modelling	no	difficult	seldom used	Difficult
Luring	difficult	difficult	good for simple actions	seldom used
Capturing	no	no	good	good
Imitating	yes	yes	seldom used	Difficult
Shaping	yes	yes	very good	not used yet

Table 1: Methods for teaching actions.

slowly until a full spin of the body is achieved. The treat is given at this stage.

- Adding the command word : The word is said only once the animal has learned the desired behavior. The trainer needs to say the command just after or just before the animal performs the behavior.
- Testing the behavior : Then the learned behavior needs to be tested and refined. The trainer clicks and treats only when the exact behavior is performed.

It is important to note that as clicker training is used for guiding the animal towards a sequence of behaviors, it can be used for two purposes : (1) to learn an unusual behavior that the animal hardly ever performs spontaneously but also (2) to learn a sequence of behaviors. We will explore these two aspects of the method with robotic clicker training.

4 Robotic clicker training

We have tried to apply the clicker training method for teaching complex actions to an enhanced version of AIBO, Sony's four-legged autonomous robot. It is to our knowledge the first time that such a method is used for robotics. Some experiments of clicker training sessions with a virtual character on a screen can be found in (Yoon et al., 2000). We will discuss the differences with our system in the last section.

Our robot has a very large set of high level preprogrammed behaviors. These behaviors range from simple motor skills (in the sense of Blumberg (Blumberg, 1997)) like walking or digging to integrated behaviors like chasing a ball involving both sensory inputs and motor skills. In its regular autonomous mode, the robot switches between these behaviors according to the evolution of its internal drives and of the opportunities offered by the environment. Some behaviors are commonly performed (e.g. chasing and kicking the ball), others are almost never observed (e.g. the robot can perform some special dances or do some gymnastic moves). More details about the execution of the normal autonomous behavior can be found in (Fujita et al., 2001).

The purpose of the present model is not to create completely new behaviors, but to have the robot performing very unusual behaviors among the set or sequences of behaviors. Clicker training can be used to guide the robot. We have decided to keep the original autonomous behavior of the AIBO and build our system "on top of it". The system acts as a "cognitive layer" which interferes with the current autonomous behavior, without controlling it completely (see also previous experiments of this sort in (Kaplan, 2000b)). The interactions of these two subsystems produce a behavior which is a mix of "innate" and learned components.

The first prototype of this system uses an external computer to perform all the additional computations concerning the cognitive layer. The computer implements speech recognition facilities which enables interactions using real words. The computer also implements a protocol for sending and receiving data between the computer and the robot through a radio connection.

Our system implements a schema-based action selection mechanism. The competition between the schemata we used in this first prototype share some similarity with the behavior action systems described in (Breazeal, 2000) or (Blumberg, 1997). Each schema is constituted by a set of activation conditions, also called releasers, and a set of actions to execute. Activation condititions are tests on sensors (e.g. presence/absence of an object, detection of word, etc.) and on previously executed schemata (e.g. schema X has been executed within last 5 seconds). The number of activation conditions fulfilled after the situation assessment determines the activation level of the schema. The schema with the highest activation level is selected. Clicker training can be implemented using 6 different schemata of this kind. We will now describe them.

4.1 Schema 1 : charging up the clicker

Before teaching new behavior, the trainer needs to "charge the clicker" :

```
TRAINER scratches the robot's head and
   says "Good"
ROBOT learns association (schema 1)
TRAINER scratches the robot's head
   again and says "Good"
ROBOT learns association (schema 1)
```

The purpose of this first schema is to learn one or several secondary reinforcers (the clicker). This schema is activated when a signal is perceived in association with a primary reinforcer. What are primary reinforcers for the robot ? It can be argued that any event fulfulling some of the robot drives can be considered as a "natural" primary reinforcer. In practice, the choice of the primary reinforcer is mainly arbitrary. In our system, we choose two stimuli to act as a primary reinforcer : detection of the sensor pressure on the robot head (giving a pat) and detection of a strong vocal congratulation (in the experiments, we use the utterance "Bravo !" which is easily distinguishable).

Each time a primary reinforcer is detected, the system looks in memory for events that occured within the last 5 seconds. These events will be potential candidates for secondary reinforcers. Theoretically these reinforcers can be anything ranging from a particular visual stimulus (detection of a special object in the image) to a vocal utterance.

As soon as the association of a given event has been detected more than 30 times (we choose the same range of training examples than the one observed for real animals), the event becomes a secondary reinforcer. The trainer can know that the event is recognized as such because the robot displays an "happy" signal (in our case wagging the tail).

With this method, it is possible to train the robot to detect several secondary reinforcers. For the experiment, we trained the system with the utterance "Good".

4.2 Schema 2 : selecting a good behavior

The trainer wants to teach a word for a special behavior that the robot performs very rarely : *the digging behavior*. In this behavior, the robot is sitting and using its left front paw to scratch the ground. Its head looks down at its paw and follows the movement. To do so, the trainer will "guide" the robot using the word "Good". *Schemata 2* and *3* are involved for this process (see figure 1).

Schema 2 is used to "select" a behavior which appears to be in the "right" direction. The schema is activated if one of the learned secondary reinforcers is detected. The system looks for the last behavior produced by the robot. If no behavior was produced during the last 10 seconds (which is very rare), nothing happens. In the other case, the behavior is selected and will be the basis for future variations.

It is important to note that, with this schema-based implementation of clicker training, the training can start at any time. There is no need for a special signal meaning "Be careful, we start the training session now". If, by chance, the robot performs a behavior that is close to the one the trainer wants to teach, the trainer can take this opportunity and start to guide the robot by saying "Good".

4.3 Schema 3 : trying a similar behavior

Schema 3 is the most important of all the training process. It spontaneously gets active when a behavior has been "selected" by schema 2. Its activation lasts for one minute. Its purpose is to try behaviors that are similar to the one previously selected. Every 20 seconds a new behavior is tried.

The difficulty is to define the notion of a "similar" behavior. To do so, a "topology" of the robot behaviors must be defined.

In the experiment we built a simple topology in which "similarity" is only defined in terms of postures. For instance two behaviors starting from a sitting position are more related than a behavior for which the robot is lying on the floor and another one for which it is standing. In the same way, two behaviors in which the robot is sitting and using its right front leg are even more similar. This kind of "descriptive" taxonomy of the behaviors is not difficult to build. It actually exists implicitly or explicitly in most robots.

But we should note that this does not take into account the potential similarity of two behaviors *in function*. For instance kicking the ball can be done both while sitting and while walking. These two similar behaviors will be classified as being very different in a topology only based on postures description. We will discuss how to improve this aspect in the last section.

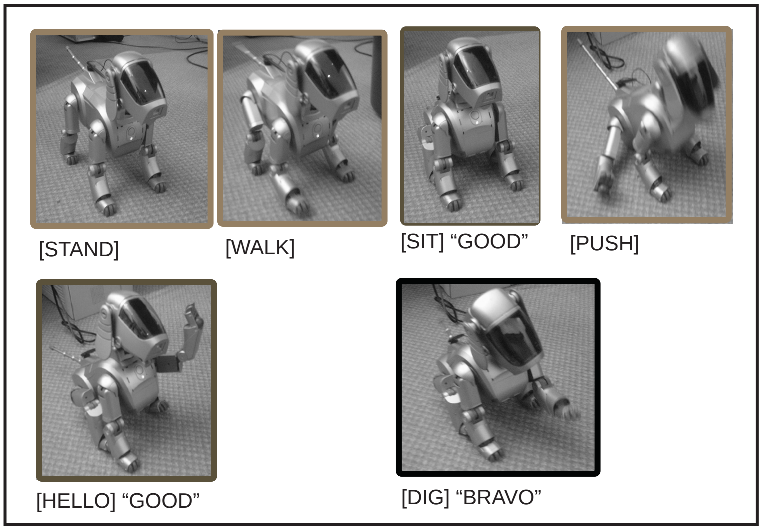

Figure 1: By using the secondary reinforcer "good" at the right time the trainer can guide the robot towards performing the desired behavior.

Our model implements the topology of the behaviors using a probabilistic graph specifying the possible transitions between each behavior (see figure 2) A behavior selected by *schema 2* corresponds to a node in this graph. *Schema 3* takes care of producing variations starting from this node using the transitions in the graph.

In our example, the trainer wants to direct the robot towards node [DIG].

```
ROBOT stands and starts walking for a
   few seconds
then ROBOT sits
TRAINER says "Good"
ROBOT selects node [SIT] (schema 2)
```

Now, the robot will try behaviors connected to the node [SIT]

```
ROBOT tries [PUSH] (schema 3) :  It
   starts pushing with its two front
   legs
TRAINER says nothing
ROBOT tries [HELLO] (schema 3) :  It
   waves its front left leg
TRAINER says "Good"
ROBOT selects node [HELLO] (schema 2)
```

This can go on as long as schema 4 is not activated, or no feedback is given to the robot for one minute.

4.4 Schema 4 : capturing the right behavior

Schema 4 is very similar to *schema 2* except that it is activated by primary reinforcers. The most recent behavior is "captured" as being a behavior that the trainer desired.

```
ROBOT tries [DIG] (schema 3) :  It
   starts digging
TRAINER says "Bravo !"
ROBOT captures [DIG] (schema 4)
```

In this case, the behavior was produced as the result of a guided search involving *schema 2* and *schema 3*. But sometimes *schema 4* can be used directly to capture a behavior that the robot has performed spontaneously.

4.5 Schema 5 : adding the command word

```
TRAINER says "Dig"
ROBOT associates "dig" with the
   sequence ([SIT] [HELLO] [DIG])
   (schema 5) and blinks its eyes
```

If a word is heard and a behavior has been recently "captured", schema 5 gets active. But in order to be sure that the right word has been understood some kind of feedback is needed. It can take several forms. If the robot is able to speak, the robot can repeat the command and asks for confirmation. If it cannot speak it must show that it has understood something for instance by blinking its eyes.

The command word is not simply associated with the last command but also with all the phases in the route towards the command, that have been marked as good steps by the emission of a secondary reinforcer. At this stage, the robot does not know

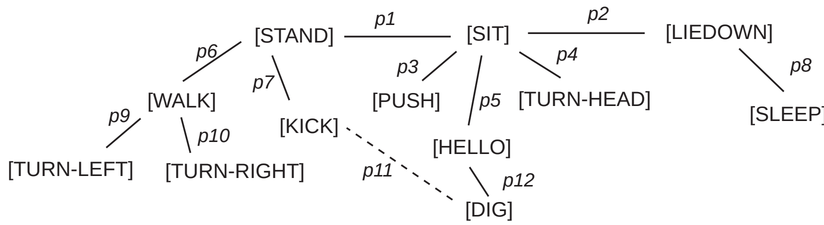

Figure 2: Part of the behavior graph for the training of the robot.

whether the word refers to the last behavior or a sequence of actions it has performed. *Schema 6* can be used to clarify this point.

4.6 Schema 6 : testing the learning

```
Later TRAINER says "Dig"
ROBOT performs the sequence ([SIT]
   [HELLO] [DIG]) (schema 6) :  It
   sits, waves its left arm and then
   starts digging
TRAINER says nothing and after a while
   repeats "Dig"
ROBOT performs the modified sequence
   ([SIT] [DIG]) (schema 6)
TRAINER says "Bravo".
```

When a word corresponding to a previously learned command is heard, *schema 6* gets active. The robot then tries to execute the corresponding sequence of behaviors. After it performs the sequence, if the robot perceives a primary reinforcer, it will consider that the command refers to the whole sequence. If not, it will associate the command heard with a new sequence, derived from the former one but involving less steps.

Each time that the command is reiterated and that *schema 6* gets active, the same kind of testing is performed. Eventually, the robot may end up considering that the command word only refers to the last action.

5 Discussions

5.1 Related works

Many aspects of the prototype we present in this paper are not new. There is already an important litterature on associative learning for virtual creatures and robots (see in particular (Blumberg, 1997) and (Billard, 1998)). It can be argued that the techniques we use for learning associations are less elaborate than the ones presented by these authors. We already mentionned that at least one paper (Yoon et al., 2000) also takes clicker training as an inspiration for training a virtual dog. In their set up, clicker training is mainly use to replace the primary reinforcer (food) by a learned clicker sound which is quicker and easier to detect. On the contrary, we use clicker training for shaping : guiding the robot towards a desired behavior. This usage, which in a simple manner enables to teach both rare behaviors and sequences of behaviors to the robot is the main originality of the present work.

5.2 Observations

This first prototype of robotic clicker training gave encouraging results during the informal tests that we have performed at the Sony CSL laboratory. Although the training technique is not ”natural” for people who are not used to dog training, it appeared to be quite easy to understand and to apply. Once the method is understood, training the robot can really be experienced as a game. Implicit competitions appear between robot owners : each one is trying to impress the others with the very rare and funny behaviors that he was able to teach. The owners can use the word they taught to their robot to discover new behaviors more quickly : ”DIG” can be the starting point of a new and even more unusual behavior.

5.3 Limitations

The main limitation of this first prototype is the fact that it relies on the definition of a good topology for the robot behaviors. Usually, the trainer is not the robot designer. For having efficient clickert training sessions, the topology needs to match with the particular way the trainer perceives whether an action is going in the right direction or not.

5.4 Improvements of the model

We made two improvements of the model to deal with this problem. As we mentionned, the initial graph is designed according to objective postural similarity. To match with the trainer's representation of the possible behavior transitions, the behavior graph needs to be adaptative. We achieve this by two means.

- Modification of the transition probability : When the user says "good" for some transitions, the probability of doing these transitions in the future increases. This way, transitions that the trainer views as natural tends to be repeated in the future.

- Creation of new archs in the behavior graph : When a sequence is learned (activation of schema 5) direct archs connecting each node of the sequence are created, initially with a low probability. For instance, it can happen that a user who wants to guide the robot towards the [DIG] behavior, initially reinforces the [KICK] behavior thinking it is going in the right direction. But in the initial graph the robot could not go directly from [KICK] to [DIG]. After several minutes, the robot finally performs the [DIG] behavior. The trainer congratulates it and teach it the word "dig". The word will be probably associated to a long sequence like for instance (KICK SIT HELLO DIG). Several new archs will be created, and among them, the direct arch KICK-DIG (see graph 2).

With these two improvements of the model, the robot tends to do more often the transitions that the trainer likes or finds natural. The robot adapts itself to be trained more easily by its teacher.

6 Conclusions

Clicker training seems to be a promising technique to teach unusual behaviors and sequences of actions to a robot. It appears to be specially well adapted to animal-like autonomous robots. It is rather easy to implement and needs small computational power. We believe that it could also be used for other purposes than teaching actions. The problem of reaching shared attention is crucial for teaching names of objects (Kaplan, 2000b; Fujita et al., 2001). A clicker training approach could be tried : the robot would be guided towards the subject of the interaction by the emission of secondary reinforcers as it is getting closer to it. If a simple solution like clicker training could facilitate such a hard robotic problem as mutual attention, it would show once again how fruitful interactions between ethologists and robot researchers can be.

7 Acknowledgements

We thank Luc Steels and Angus McIntyre for their precious comments. We are also indebted to anonymous reviewers for remarks that have led to improvements of this text. This work has been partly supported by OTKA (T 029705) and by a grant (F 226/98) from the Hungarian Academy of Sciences.

References

Arkin, R. (1998). *Behavior-Based Robotics*. MIT Press, Cambridge, MA.

Bakker, P. and Kuniyoshi, Y. (1996). Robot see, robot do : An overview of robot imitation. In *Proceedings of AISB Workshop on Learning in Robots and Animals*, pages 3–11, Brighton, UK.

Billard, A. (1998). *DRAMA, a connectionist model for robot learning : Experiments in grounding communication through imitation in autonomous robots*. PhD thesis, University of Edinburgh.

Billard, A., Dautenhahn, K., and Hayes, G. (1998). Experiments on human-robot communication with robota, an interactive learning and communicating doll robot. In Edmonds, B. and Dautenhan, K., editors, *Socially situated intelligence workshop (SAB 98)*, pages 4–16.

Blumberg, B. (1997). *Old tricks, new dogs : Ethology and interactive creatures*. PhD thesis, MIT Media Laboratory.

Blumberg, B., Tood, P., and Maes, P. (1996). No bad dogs : Ethological lessons for learning in hamsterdam. In Maes, P., M.J. Mataric, M., Meyer, J.-A., Pollack, J., and Wilson, S., editors, *From Animals to Animats, Proceedings of the Fourth International Conference on the Simulation of Adaptive Behavior*, pages 295–304, Cambridge, MA. MIT Press/Bradford Books.

Breazeal, C. (2000). *Sociable machines : expressive social exchange between humans and robots*. PhD thesis, MIT.

Dautenhahn, K. (1995). Getting to know each other : artificial social intelligence for autonomous robots. *Robotics and autonomous systems*, 16:333–356.

Dautenhahn, K. (1999). Embodiment and interaction in socially intelligent life-like agents. In Nehaniv, C., editor, *Computation for metaphors, analogy and agent*, volume 1562 of *Springer Lecture Notes in Artificial Intelligence*, pages 102–142. Springer.

Druin, A. and Hendler, J. (2000). *Robots for kids : Exploring new technologies for leaning*. Morgan Kaufman Publishers.

Fujita, M., Costa, G., Takagi, T., Hasegawa, R., Yokono, J., and Shimomura, H. (2001). Experimental results of emotionally grounded symbol acquisition by four-legged robot. In Muller, J., editor, *Proceedings of Autonomous Agents 2001*.

Fujita, M. and Kitano, H. (1998). Development of an autonomous quadruped robot for robot entertainment. *Autonomous Robots*, 5.

Kaplan, F. (2000a). Free creatures: The role of uselessness in the design of artificial pets. In Christaller, T., Indiveri, G., and Poigne, A., editors, *Proceedings of the 1st Edutainment Robotics Workshop*. GMD-AiS.

Kaplan, F. (2000b). Talking aibo : First experimentation of verbal interactions with an autonomous four-legged robot. In Nijholt, A., Heylen, D., and Jokinen, K., editors, *Learning to Behave: Interacting agents CELE-TWENTE Workshop on Language Technology*, pages 57–63.

Kaplan, F. (2001). Un robot peut-il être notre ami ? In Orlarey, Y., editor, *L'Art, la pensée, les émotions*, pages 99–106. Grame.

Kuniyoshi, Y., Inaba, M., and Inoue, H. (1994). Learning by watching : Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6):799–822.

Kusahara, M. (2000). The art of creating subjective reality: an analysis of japanese digital pets. In C., M. and Boudreau, E., editors, *Artificial life VII Workshop Proceedings*, pages 141–144.

Pryor, K. (1999). *Clicker training for dogs*. Sunshine books, Inc., Waltham, MA.

Roy, D. (1999). *Learning form sights and sounds : a computational model*. PhD thesis, MIT Media Laboratory.

Skinner, B. (1938). *The Behavior of Organisms*. Appleton Century Crofs, New York, NY.

Steels, L. and Brooks, R. (1994). *The 'artificial life' route to 'artificial intelligence'. Building Situated Embodied Agents*. Lawrence Erlbaum Ass, New Haven.

Tillman, P. (2000). *Clicking with your dog*. Sunshine Books, Walthman, MA.

Webb, B. (2000). What does robotics offer animal behaviour ? *Animal behavior*, 60:545–558.

Yoon, S.-Y., Burke, R., Blumberg, B., and Schneider, G. (2000). Interactive training for synthetic characters. In *AAAI /IAAI 2000*, pages 249–254.