Robotic clicker training

Frédéric Kaplan [a],*, Pierre-Yves Oudeyer [a], Enikö Kubinyi [b], Adám Miklósi [b]

[a] *Sony CSL Paris, 6 Rue Amyot, 75005 Paris, France*
[b] *Department of Ethology, Eötvös University, Göd, Hungary*

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. © 2002 Published by Elsevier Science B.V.

Keywords: Robot training; Dog training techniques; Pet robots

1. Introduction

The recent years have been characterized by the expansion of animal-like entertainment robots [1–3]. The AIBO, commercialized by Sony in 1999, was the first product of this new generation of robots [4]. 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 perform 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 [3].

One of the challenges and pleasures in keeping a real pet, like a dog, is that the owner has to train 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 inter-

esting relationship to emerge between an entertainment robot and its owner (Dautenhahn [5] develops a similar argument). For this reason, a growing number of research groups are currently focusing on teaching techniques for autonomous robots [6–10].

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 [11–13] for instance). But apart from the exception of Blumberg's team at MIT Media Lab [14–16], the field of animal training has not yet been much investigated as a source of inspiration for robotics researchers.

How to teach complex behaviors (and possibly commands associated to them) only through interactions

* Corresponding author.
E-mail address: kaplan@csl.sony.fr (F. Kaplan).

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 [17,18]. 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 section describes our implementation of a first prototype of a training session with AIBO, Sony's four-legged robot. The last section 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 mention-

ing 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 that are efficient at sharing attention and discriminating stimuli is very difficult, it is advisable to operate in 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 *modeling* method (the term "molding" is also used), often tried by dog owners, is almost never used by professional trainers. It involves physically manipulating 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. Modeling has been mainly used to teach positions to robots in industrial contexts. As soon as

Table 1
Methods for teaching actions

Training techniques	Sequences of actions	Unusual actions	Usability with animals	Usability for autonomous robots
Modeling	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	Seldom used

the robot is autonomous and constantly active, its manipulation becomes problematic. Only partial modeling 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 modeling 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 automatically 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 [19] 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 modeling and luring, *capturing* methods exploit behaviors that the animal performs spontaneously. For instance, every time a dog owner acknowledges 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 [5]). 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 5 years (see for instance [6,20–22]). 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 into 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. Shaping has been seldom said in robotics, one exception being the work of Dorigo and Colombetti [23]. To our knowledge, this paper presents the first application of clicker training for shaping the behavior of an autonomous entertainment robot.

3. A brief introduction to clicker training

Clicker training is based on Skinner's theory [24] of operant conditioning. In the 1980s Gary Wilkes, a behaviorist, collaborated with Karen Pryor, a dolphin trainer, to popularize this method for dog training [17]. 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 *conditioned 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 criterion is raised 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 [16]. 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 [15]) 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 [4,10,25].

In our system, the behavior of the robot is implemented through a schema-based action selection mechanism. The competition between the schemata we used in this first prototype shares some similarity with the behavior action systems described in [26] or [15]. Each schema is constituted by a set of activation conditions (defining an activation level), also called releasers, and a set of actions to execute. The activation level depends on sensors state (e.g. presence/absence of an object, detection of word, etc.), on previously activated schemata (e.g. schema X has been executed within last 5 seconds) and on the state of three independent modules (emotion, instincts and user's expectations). [1] The schema with the highest activation level is selected.

The schemata are organized under hierarchical trees. At the root of the tree, schemata are defined in terms of general goals. A high-level schema can be for instance dedicated for approaching a detected object such as the red ball. The actions defined in this schema specify how to test whether the approach has succeeded but not the way the robot should do it. For that part a new competition between schemata of the tree subnodes (running, walking, etc.) will occur, and so on.

---

[1] The first prototype of this system uses an external computer to perform all the additional computations concerning the training module. 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.
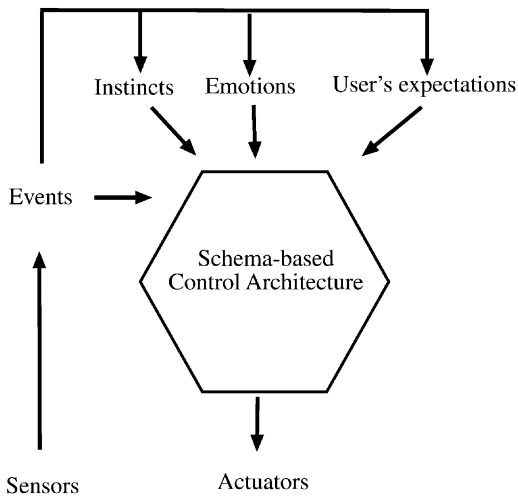
Fig. 1. A simplified view of the control architecture of our proto-type.

Clicker training is used to guide the robot. It can be used for two purposes: (1) to create new behaviors by combining existing ones; or (2) to have the robot performing very unusual behaviors among the set. It is through the effects of the user's expectation module that the behavior of the robots is modified to fit with the training performed by the user. With this new model, the behavior of the robot results of the combination of three independent forces: opportunities in the environments, natural instincts and emotions of the robot, and models of the user's expectation at the time of the action (Fig. 1).

4.1. Learning secondary reinforcers

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

```
TRAINER scratches the robot's head
  and says ''Good''
ROBOT learns association in user's
  expectation module
TRAINER scratches the robot's head
  again and says ''Good''
ROBOT learns association in user's
  expectation module
```

The user's expectation module includes an associative memory capable of detecting and record-ing associations between primary reinforcers and words/sounds. 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 occurred 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 as 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. Guiding the robot

Let us assume that the trainer wants to teach the robot to spin in a clockwise circle. This is a thing that some dog owners do with clicker training as we already mentioned. First, the trainer needs to inform the robot that he is waiting for the performance of a special action. In our experiment, we use the detection of the special word "try". Detecting this word, the user's expectation module becomes active.[2] Its purpose is to build a model of the behavior that the trainer desires.

In the beginning, as no model is available, the action selection mechanism is working as in the normal autonomous mode. At the bottom of the schemata tree, high-level schemata are competing (Fig. 2). For instance, the *play* schema may be selected because

---

[2] It is not difficult to make the system more flexible. The trainer can for instance opportunistically say "Good" to capture a behavior that the robot has performed spontaneously, provoking at the same time the activation of the user's expectation module.
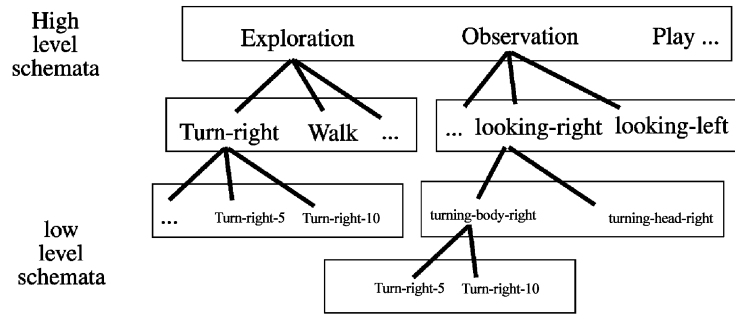
Fig. 2. Partial view of the schemata hierarchical trees.

of the presence of the red ball and of the high *curiosity* motivation level in the instinct module. The user's expectation module is active but not detecting any secondary reinforcer, so it starts inhibiting the current play schema, forcing the robot to try a new one. Eventually, a behavior like walking is exhibited under the high-level *exploration* schema. The trainer says "Good" and the training module marks the top level exploration schema as a first model of what the trainer wants. It also memorizes which particular low-level schema was reinforced. One immediate effect is to reinforce the exploration schema during the high-level schemata selection phase, favoring the execution of subschemata such as walking, turning, etc. The user's expectation module progressively inhibits the subschemata that are not correlated with any reinforcement. When the robot turns right for the first time, the trainer says "Good" and the process is reiterated.

Let us imagine that the robot ends up by performing a particular *turn-right* schema using a 10-step movement that the trainer will reinforce systematically. As no additional variation on the schema is possible, the robot will simply perform it over and over again. At some point it will have done the expected clockwise circle and the trainer will say "Bravo".

The same results could have been obtained taking another route. The *right-turn* schema is also a subschema of the *looking right* higher-level schema, itself governed by the *observation* schema. If the robot had turned its head to the right, the trainer could have reinforced that behavior and encouraged the robot to test other strategies under in the same schema tree.

This guiding process exploits the fact that the behaviors of the robot are organized in a hierarchical manner. As a consequence, the user's expectation module induces first the general expectation of the trainer (high-level schema) and then its actual instantiation into a particular behavior.

4.3. Adding the command word

```
TRAINER says ``spin''
ROBOT associates ``spin'' with the
  sequence
(WALK-4, TURN-LEFT-3, TURN-RIGHT-5,
  TURN-RIGHT-10, TURN-RIGHT-10,
TURN-RIGHT-10, TURN-RIGHT-10) and
  blinks its eyes.
```

If a word is heard [3] and the user's expectation module has a complete new action in memory, it will associate both. The word is not simply associated with the last schema executed but with all the schemata reinforced during the training session (that have been marked as good steps by the emission of a secondary reinforcer). At this stage, the robot does not know whether the word refers to the last schema or to a sequence of actions it has performed. So the system creates a temporary schema which has the word as an activation condition and the sequence as an action list. This temporary schema is added as a subschema of

---

[3] 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 ask for confirmation. If it cannot speak it must show that it has understood something, for instance by blinking its eyes.

the high-level *pleasing-user* schema. As we will see temporary schemata are associated with a confidence level that increases when the robot perceives reinforcement just after their execution. After reaching a given confidence level a temporary schema can become a standard one.

4.4. Testing the newly created schema

```
Later, TRAINER says ``spin''
ROBOT performs the sequence
(WALK-4, TURN-LEFT-3, TURN-RIGHT-5,
  TURN-RIGHT-10, TURN-RIGHT-10,
TURN-RIGHT-10, TURN-RIGHT-10)
It walks forward, makes three steps in
  the left direction and then moves to
  the right in a chaotic manner.
TRAINER says nothing and after a while
  repeats ``spin''
ROBOT performs the modified sequence
  (WALK-4, TURN-RIGHT-15
TURN-RIGHT-10, TURN-RIGHT-10,
  TURN-RIGHT-10), etc.
```

When a word corresponding to a previously learned command is heard, the *pleasing-user* high-level schema should get active as well as the dedicated temporary subschema. The robot tries to execute the sequence of schemata corresponding to the action part of the temporary schema. If the robot perceives a primary reinforcer after it performs the sequence, it will consider that the command referred to the whole sequence and will increase its confidence in it. If not, it will derive a new sequence from the existing one by applying randomly a series of operators. These operators include pruning (suppressing a schema in the sequence at random), factorization (merging elements of the sequence such as transforming (TURN-RIGHT-5, TURN-RIGHT-10) into (TURN-RIGHT-15) and mutation (changing one element by a similar one, similarity being defined by the position of the two schemata in the hierarchical trees).

This kind of testing can go on for sometime (but may be suspended and continued later on), until the trainer is really satisfied with the robot behavior. Eventually the new schema is added into the normal behavior of the robot.

5. Discussions

5.1. Related work

Some aspects of the prototype we present in this paper are not new. There is already an important literature on associative learning for virtual creatures and robots (see in particular [15,19]). It can be argued that the techniques we use for learning associations are less elaborate than the ones presented by these authors. We already mentioned that at least one paper [16] also takes clicker training as an inspiration for training a virtual dog. In their set up, clicker training is mainly used to replace the primary reinforcer (food) by a learned clicker sound which is quicker and easier to detect. 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. Experiments, observations and improvements of the model

We are currently conducting experimental tests with "naive" users to evaluate the ease of use of this first prototype (Fig. 3 shows a typical session). The candidates, aged between 15 and 25 years have never interacted with an autonomous robot before. They are instructed that this robot can be trained using the words "Good" and "Bravo" (the robot is previously trained with these words before the test sessions). The sessions are videotaped and analyzed in collaboration with an ethologist specialist in human–dog interactions.

Although the precise analysis of these sessions are still undergoing, we can already make several general observations.

- Difficulties come more from the quality of the speech recognition than from the method itself. 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.
- Users are sometimes surprised by the effects of their training. They would not have expected that reinforcing action A will lead to action B. This problem is linked to the definition of a good schema-hierarchy for the robot behaviors.

Fig. 3. Test sessions with naive users.

Usually, the trainer is not the robot designer. For having efficient clicker training sessions, the schema-hierarchy needs to match with the particular way the trainer perceives whether an action is going in the right direction or not.

One improvement of the model can be that the user's expectation module has an effect not only in the short-term current interactions but also in the long-term behaviors directed towards that particular user. To match with the trainer's representation of the possible behavior transitions, the schema-hierarchy needs to be adaptive. When the user says "Good" for some transitions (schema-1 to schema-2), the probability of doing these transitions in the future should increase. In practice, to increase the probability of moving from schema-1 to schema-2, an additional activation condition is added to schema-2 specifying that the recent activation of schema-1 should increase its activation level for the competition. This way, transitions that the trainer views as natural tend to be repeated in the future.

Another improvement can be that all the schemata added to the schema-hierarchy by the user's expectation module get more activation when the presence of the user is detected, even if no training session is active. It means for instance that the robot, detecting the presence of the user, will spontaneously do its spin behavior. These kind of learned user-specific behavioral responses can be greatly valuable for the robot's owner as we argued in [3].

6. Conclusion and perspective

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. More generally, it is a good example of what robotic engineers can learn from ethologists and animal trainers. We think that, in return, such a robotic model can also be interesting for ethological studies. In the near future, we wish to compare the results obtained with this first prototype to ethological studies of clicker training session with real dogs. By doing so, we will certainly learn more about the specificities of the robots, the dogs and how their behavior is interpreted in the eyes of the human observer.

Acknowledgements

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

[1] M. Kusahara, The art of creating subjective reality: An analysis of Japanese digital pets, in: M.C.E. Boudreau (Eds.), Proceedings of the VII Workshop on Artificial Life, Portland, OR, 2000, pp. 141–144.

[2] A. Druin, J. Hendler, Robots for Kids: Exploring New Technologies for Learning, Morgan Kaufman, San Mateo, CA, 2000.

[3] F. Kaplan, Artificial attachment: Will a robot ever pass Ainsworth's strange situation test? in: Proceedings of the IEEE–RAS International Conference on Humanoid Robots (Humanoids 2001), Tokyo, 2001.

[4] M. Fujita, H. Kitano, Development of an autonomous quadruped robot for robot entertainment, Autonomous Robots 5 (1) (1998) 7–18.

[5] K. Dautenhahn, Embodiment and interaction in socially intelligent life-like agents, in: C. Nehaniv (Ed.), Computation for Metaphors, Analogy and Agent, Lecture Notes in Artificial Intelligence, Vol. 1562, Springer, Berlin, 1999, pp. 102–142.

[6] A. Billard, K. Dautenhahn, G. Hayes, Experiments on human–robot communication with Robota: An imitative learning and communication doll robot, in: B. Edmonds, K. Dautenhahn (Eds.), Socially Situated Intelligence: A workshop held at SAB'98, Zurich, Switzerland, 1998, pp. 4–16.

[7] D. Roy, Learning form sights and sounds: A computational model, Ph.D. Thesis, MIT Media Laboratory, Cambridge, MA, 1999.

[8] F. Kaplan, Talking AIBO: First experimentation of verbal interactions with an autonomous four-legged robot, in: A. Nijholt, D. Heylen, K. Jokinen (Eds.), Learning to Behave: Interacting Agents, CELE-Twente Workshop on Language Technology, Enschede, Netherland, 2000, pp. 57–63.

[9] L. Steels, F. Kaplan, AIBO's first words, Evolution of Communication 4 (1) (2002).

[10] M. Fujita, G. Costa, T. Takagi, R. Hasegawa, J. Yokono, H. Shimomura, Experimental results of emotionally grounded symbol acquisition by four-legged robot, in: J. Muller (Ed.), Proceedings of Autonomous Agents 2001, Montreal, Quebec, 2001.

[11] L. Steels, R. Brooks, The 'artificial life' route to 'artificial intelligence', Building Situated Embodied Agents, Lawrence Erlbaum Ass, New Haven, CT, 1994.

[12] R. Arkin, Behavior-based Robotics, MIT Press, Cambridge, MA, 1998.

[13] B. Webb, What does robotics offer animal behaviour? Animal Behavior 60 (2000) 545–558.

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

[15] B. Blumberg, Old tricks, new dogs: Ethology and interactive creatures, Ph.D. Thesis, MIT Media Laboratory, Cambridge, MA, 1997.

[16] S.-Y. Yoon, R. Burke, B. Blumberg, G. Schneider, Interactive training for synthetic characters, in: AAAI/IAAI 2000, Austin, TX, 2000, pp. 249–254.

[17] K. Pryor, Clicker Training for Dogs, Sunshine Books, Inc., Waltham, MA, 1999.

[18] P. Tillman, Clicking with Your Dog, Sunshine Books, Waltham, MA, 2000.

[19] A. Billard, Drama, A connectionist model for robot learning: Experiments in grounding communication through imitation in autonomous robots, Ph.D. Thesis, University of Edinburgh, 1998.

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

[21] K. Dautenhahn, Getting to know each other: Artificial social intelligence for autonomous robots, Robotics and Autonomous Systems 16 (1995) 333–356.

[22] P. Bakker, Y. Kuniyoshi, Robot see, robot do: An overview of robot imitation, in: Proceedings of the AISB Workshop on Learning in Robots and Animals, Brighton, UK, 1996, pp. 3–11.

[23] M. Dorigo, M. Colombetti, Robot Shaping—An Experiment in Behavior Engineering, MIT Press, Cambridge, MA, 1998.

[24] B. Skinner, The Behavior of Organisms, Appleton Century Crofs, New York, 1938.

[25] M. Fujita, H. Kitano, T. Doi, Robot entertainment, in: A. Druin, J. Hendler (Eds.), Robots for Kids: Exploring New Technologies for Learning, Morgan Kaufmann, San Mateo, CA, 2000, pp. 37–70 (Chapter 2).

[26] C. Breazeal, Sociable machines: Expressive social exchange between humans and robots, Ph.D. Thesis, MIT Media Laboratory, Cambridge, MA, 2000.

Frédéric Kaplan is a researcher at Sony Computer Science Laboratory in Paris. He graduated as an engineer of the École Nationale Supérieure des Télécommunications in Paris and received a PhD degree in Computer Science from the University Paris VI. He joined Sony CSL in 1997, where his work focused on new techniques for human-robot interactions and on the emergence of cultural systems among machines. Recently, he developed several prototypes showing how a four-legged autonomous robot, like Sony's AIBO, can learn the meaning of words used by his master.

Pierre-Yves Oudeyer is a researcher at Sony Computer Science Laboratory in Paris. He has been conducting research projects within the areas of collective robotics, interaction protocols in multi-agent systems, and the computational modelling of the origins of language. He has recently developed techniques for the production and recognition of emotions in speech for the Sony humanoid robot SDR-5. He is the author of 7 patents concerning interaction technologies for robots.

Adám Miklósi is a research fellow at the Department of Ethology, University in Budapest. He received his PhD in Ethology studying the antipredator behaviour of the paradise fish. At present he is interested in the sociocognitive abilities of the dog, with special emphasis on dog-human communication, social learning and dog-human attachment. This research supports the hypothesis that during domestication dogs acquired behavioural traits that enhance their ability to live in the human environment. With his colleauges he has published over 10 papers on this subject. He is Associate Editor of *Animal Cognition* (Springer Verlag).

Enikö Kubinyi is a researcher at the Department of Ethology, Eötvös Loránd University in Budapest. She studies the information transmission via social learning between humans and dogs, and works in a project, which is aimed at revealing the behavioural differences between hand-raised wolves and dogs. Recently she has carried out ethological experiments with dogs encountering a four-legged autonomous robot.