Neural Networks 23 (2010) 1113–1124

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2010 Special Issue

Self discovery enables robot social cognition: Are you my teacher?

Krishnanand N. Kaipa [a], Josh C. Bongard [a,*], Andrew N. Meltzoff [b]

[a] Department of Computer Science, University of Vermont, Burlington VT, USA

[b] Institute for Learning and Brain Sciences, University of Washington, Seattle WA, USA

ARTICLE INFO

Article history:
Received 18 April 2010
Received in revised form 15 July 2010
Accepted 29 July 2010

Keywords:
Social robotics
Self discovery
Social cognition
Teacher identification
Self-other equivalence
Robot learning by imitation

ABSTRACT

Infants exploit the perception that others are 'like me' to bootstrap social cognition (Meltzoff, 2007a). This paper demonstrates how the above theory can be instantiated in a social robot that uses itself as a model to recognize structural similarities with other robots; this thereby enables the student to distinguish between appropriate and inappropriate teachers. This is accomplished by the student robot first performing self-discovery, a phase in which it uses actuation–perception relationships to infer its own structure. Second, the student models a candidate teacher using a vision-based active learning approach to create an approximate physical simulation of the teacher. Third, the student determines that the teacher is structurally similar (but not necessarily visually similar) to itself if it can find a neural controller that allows its self model (created in the first phase) to reproduce the perceived motion of the teacher model (created in the second phase). Fourth, the student uses the neural controller (created in the third phase) to move, resulting in imitation of the teacher. Results with a physical student robot and two physical robot teachers demonstrate the effectiveness of this approach. The generalizability of the proposed model allows it to be used over variations in the demonstrator: The student robot would still be able to imitate teachers of different sizes and at different distances from itself, as well as different positions in its field of view, because change in the interrelations of the teacher's body parts are used for imitation, rather than absolute geometric properties.

1. Introduction

The field of social robotics attempts to exploit insights from developmental psychology for the design of adaptive robots. This approach assumes that increasingly complex behaviors can be realized in machines by having them learn from humans or other robots automatically, rather than directly programming behaviors in. However, in order to do so a robot must autonomously identify an appropriate teacher, where the quality of a teacher requires several conditions to be met: the teacher must be located within sensor range of the student (even though it could still leave, and return to, the sensory field from time to time); it must exhibit behaviors that the student can potentially perform; and it must adaptively expand its demonstrations as the student's abilities improve. Thus a major challenge in social robotics is how best to equip a machine to identify appropriate teachers[1] in its environment. Teacher identification however raises the question of how a student machine can establish that a candidate teacher is sufficiently similar to itself such that social interaction and learning can take place.

This paper presents a novel approach to the problem of teacher identification and imitation. It is predicated on the assumption that the student does not initially have access to a model of itself, nor that of a candidate teacher. It must therefore build a self model, a model of a candidate teacher, and determine whether there is a correspondence between the two. This approach is inspired by the way infants exploit the perception that others are 'like me' as a starting point for social cognition (Meltzoff, 2007a): they recognize cross-modal equivalences between acts they see others perform and their own felt bodily movements and this recognition of self-other equivalences in action gives rise to interpreting others as having similar psychological states such as perceptions and emotions. In particular, we instantiate the above theory in a social student robot that first builds a self model, and then uses this self-model to discover self-other equivalences with other robots in its surroundings. Whether such equivalences can be found therefore enables the student to distinguish an appropriate from an inappropriate teacher. This work addresses the question of "whom to imitate", one of the open problems in social cognition and one of growing centrality for robotics and neuroscience.

The proposed framework consists of three phases: *Self discovery*, *teacher modeling*, and *teacher imitation*. First, the student performs self-discovery, a phase in which it uses actuation-perception

* Corresponding author.
E-mail addresses: kkrishna@uvm.edu (K.N. Kaipa), jbongard@uvm.edu, josh.bongard@uvm.edu (J.C. Bongard), meltzoff@u.washington.edu (A.N. Meltzoff).

1 Here, an appropriate teacher is defined as the most efficient teacher that the student can learn from.

0893-6080/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2010.07.009

relationships to infer its own mechanical structure. Second, during teacher modeling, the student models a candidate teacher using a stereo vision based active learning approach to create an approximate physical simulation of the teacher. Third, during the teacher imitation, the success of finding a neural controller that allows its self model (evolved in the first phase) to imitate the action of the teacher's visual model (evolved in the second phase) determines whether or not the observed agent is an appropriate teacher. Fourth, the student uses the neural controller (learnt in the third phase) to move, resulting in imitation of the appropriate teacher. This approach was designed such that teachers may be deemed appropriate or not even if their visual appearance differs from that of the student robot. For instance the student robot would still be able to imitate teachers of different size and at different distances from itself, as well as different positions in its field of view, as explained later. This sort of flexibility is very helpful and has many applied implications.

This approach differs from other work that address the same problem in several ways.

1.0.1. *Aspects of imitation*

Imitation is one of the forms of social learning that enables robots to learn new skills. Several important aspects of robotic imitation were summarized by Breazeal and Scassellati (2002) and Dautenhahn and Nehaniv (2002) who listed the issue of "whom to imitate" as being one of the key unsolved problems. Most of the work on imitation-based learning has sidestepped this problem by defining a fixed role for one agent as being the demonstrator and another agent as being the imitator (Jansen & Belpaeme, 2006). To the best of our knowledge, our work is one of the first to address the issue head on. In particular, the social robotics framework proposed here offers a robot design that includes the capacity to observe actions of candidate agents and determine which agent will be a good teacher that can and should be imitated.

1.0.2. *Self-other equivalence*

In most robotic imitation-based learning schemes the student attempts to find a correspondence between the actions of the teacher and its own behavior repertoire; the teacher's body structure could range from being identical (Jansen & Belpaeme, 2006) to similar and dissimilar (Alissandrakis, Nehaniv, & Dautenhahn, 2007; Coates, Abbeel, & Ng, 2008). However, in all these research attempts the student assumes that it knows the model of its body structure *a priori*. That is, it knows beforehand how its body parts are attached to each other; the forward kinematics – the mapping from the joint space to the end effector space – is also known *a priori* in most cases. However, in our approach, we relax that assumption: the student does not know about its own structure *a priori*; instead, it performs self-discovery and discovers its own self-model. Importantly, this self-model shapes its perceptions of candidate teachers: if the learned self-model is composed of k body parts, the student will attempt to model candidate teachers using the same number of body parts. This in essence helps it to 'filter out' entities that would be poor teachers, such as those with sufficiently different mechanical structures. Once a teacher model has been created, the student robot actuates its self-model in an attempt to establish a correspondence between the perceived motions of the modeled teacher.

Within developmental psychology theory, there are opposing hypotheses about the origins of self-concepts in infants. One view holds that the young infant learns about itself primarily and at first through interaction with others (Cooley, 1902; Mead, 1934). Infants have no prior self-concepts and learn about the possibilities and powers of their own actions through observing the reactions of others to their behavior. On the other side, theorists argue that primitive self-concepts exist prior to such social experience (Butterworth, 1992; Meltzoff, 2007a). In this view, infants have a proprioceptive sense of self that derives in part from their own body movements, which Meltzoff and Moore (1997) have called "body babbling". In this view a primitive sense of self is the precondition for successful social interaction rather than an outcome of it. Infants come to understand others because they are perceived to be "like me" and the behavior of others is imbued with felt meaning in part based on the child's own prior self-experiences (Meltzoff, 2007b). In the approach adopted here, the student robot figures out a model of its self to some extent by self-directed exploration and then uses the discovered self-model to find out a match with other similar robots that would be candidates as effective teachers. Our proposal contributes to knowledge in two ways. It uses computer and robotic modeling to rigorously investigate the feasibility of the hypothesis that self-exploration could be a foundational step in developing social cognition. Reciprocally, it exploits this developmental psychology theory to help us design a robot that can learn from observing and imitating experts in the surrounding culture. An improved understanding of imitation has broad implications for the science of learning both in man and machine (Meltzoff, Kuhl, Movellan, & Sejnowski, 2009).

The paper is organized as follows. Prior work on robots learning by imitation is described in Section 2. The proposed framework and the methods used are described in Section 3, followed by results in Section 4 and some discussion in Section 5. Conclusions are presented in Section 6.

2. Prior work on imitative learning in robots

Robot imitation requires complex mechanisms that map the observed movement of a teacher (such as another robot or a human) onto its own movement repertoire. Several researchers have addressed different aspects of this challenge, known as the correspondence problem.

Approaches are diverse: The agents used to play the roles of student and teacher range from anthropomorphic robot arms (Jansen & Belpaeme, 2006; Schaal, 1997) to simulated robots (Amit & Mataric, 2002), as well as humanoid robots (Billard & Schaal, 2001) with redundant degrees of freedom. The nature of imitation varied from reproducing the demonstrator's action (Amit & Mataric, 2002; Billard & Schaal, 2001) to performing an action that achieves the underlying goal of the observed action (Jansen & Belpaeme, 2006; Shon, Storz, Meltzoff, & Rao, 2007), and learning correctly from a 'flawed' demonstration (Breazeal, Berlin, Brooks, Gray, & Thomaz, 2006). In some cases, the student focussed on observing either the teacher's bodily movements (Amit & Mataric, 2002; Billard & Schaal, 2001) or the effect of the teacher's interaction with the environment (Jansen & Belpaeme, 2006; Schaal, 1997); accordingly, the observation module ranged from special data acquisition systems to vision-based acquisition systems.

Learning from demonstrations in robotics is inspired by the way humans learn from instructions and/or demonstrations by other humans. Schaal (1997) showed how demonstration can accelerate learning in the context of reinforcement learning-based control: an anthropomorphic robot arm learned to balance a pole in just a single trial by observing a 30 s human demonstration. The robot was equipped with a 60 Hz video-based stereo vision system, and the pole was marked with two colors which could be tracked in real time. The input to the student imitation module was only the movements of the pole as a function of time: that is, the effect of the teacher's interaction with the pole.

Jansen and Belpaeme (2006) focussed on the problem of how an agent can know what aspect of the observed teacher's behavior it should imitate. They proposed a model which allowed a student to interpret the demonstrated behavior through repeated interactions. The student and teacher were simulated robot arms with identical body configurations; each agent used its end effector to pick and place three colored blocks on a 2D board comprising 5×5 squares. The authors assumed that the teacher and student both had the same set of actions with which to manipulate their environment. The student did not observe the action of the teacher; instead, it observed the effect of the teacher's actions on the blocks. Therefore, the student has to resolve a correspondence between the end effect achieved by their respective action sequences rather than a correspondence between their bodily movements.

Billard and Schaal (2001) used artificial neural networks (ANNs) for robot learning by demonstration. A simulated 41-DOF humanoid implemented the ANN model to reproduce the motion of a human arm. The demonstrated behavior consisted of left arm movements for five repetitions of drawing a figure eight. The data was gathered by a sensor suit that recorded the positions of 35 degrees of freedom of the human body. Therefore, in this work the student attempted to imitate the bodily movements of the teacher.

Amit and Mataric (2002) used a hierarchy of three computational layers – base primitives, movement specializers, and sequence learners – to learn movement sequences from demonstration. First, each base primitive was hardwired and used a parameterized encoding of motor programs to execute a class of movements; different movements of the same class could be obtained by changing the parameters. The base primitives had a visuo-motor property in that they could ascertain the values of the parameters resulting in the execution of known movements that closely approximated the observed movements. However, the fixed nature of the base primitive layer imposed constraints on the learnable movements of the system. Second, the movement specializers used information from base primitives to learn specialized movements. For example, a base primitive for reaching movements by the right hand was used by an associated movement specializer to learn specific reaches, such as reaching the nose with the same hand. Third, the sequence learners encoded a probabilistic ordering over the set of movement specializers in order to learn composite movement sequences. The authors assumed that both the demonstrator and the imitator had the same kinematic limb structure, and had similar degrees of freedom in the corresponding joints. A physics-based humanoid simulation was used to perform a series of learning trials using synthetic and human motion capture data obtained by a marker-based capture system. The human demonstrated movements consisted of reaching to various positions, painting-like movements, and drawing simple figures. Synthetic motion consisted of aerobic-style movements like stretching the arm sideways, or taking it to a vertically upright position. Similar to the work of Billard and Schaal, the bodily movements of the demonstrator were successfully imitated by the student.

Note that in all the above studies, the student learned from a single teacher, with the assumption that the teacher was imitatable. In addition, the student is equipped with a self image *a priori*; the forward kinematics, that is the mapping from actuation space to the end effector space, is known beforehand.

Lopes and Santos-Victor (2007) present a strategy for robotic imitation using a learning pathway comprising three levels: (i) Learning about the self; (ii) Learning about objects and the world and (iii) Learning about others and imitation. In the first level, the robot learns a sensory-motor map (SMM) by self-exploration that allows itself to acquire capabilities like vergence control, object foveation and perception–action coordination. In the second level, the robot builds a map of the surrounding area (object positions and identification), studies objects, their properties and how they are used by others (e.g., develops a closed-loop control behavior capable of grasping). In the third level, the robot is able to look at gestures and repeat them. Vision based rectangle-template matching is used to model the arm position of the demonstrator.

The self-exploration aspect in this work is similar to the self-discovery performed by the student in our proposed framework. However, it differs from our approach in two important aspects: First, the student robot in Lopes and Victor's study learns a correspondence between motor actions and camera images that allow reaching for objects in easy positions. However, it does not obtain a self-model of its morphology (how its body parts are attached to each other). Second, the student robot does not determine whether or not the teacher has an embodiment that is similar to its own embodiment, as the student robot described in this paper does.

Alissandrakis et al. (2007) address the problem of body mapping in robotic imitation where the demonstrator and imitator may not share the same embodiment (such as differing DOFs, body morphologies, constraints, and/or affordances). They use *correspondence matrices* to characterize partial, mirror symmetric, one-to-one, one-to-many, many-to-one, and many-to-many associations between various DOFs across dissimilar embodiments. The authors show how metrics for behavior matching can be mathematically determined by such correspondence mappings, which may serve to guide a robot regarding how to imitate an observed demonstration. However, the validation of their approach is limited to simulated robots. Also, the student assumes that its kinematic model is known *a priori*.

Nyuwa, Katagami, and Nitta (2008) propose a method that allows a robot to learn from multiple instructors. In particular, each one of 40 human participants taught a commercial robot (Sony's AIBO robot dog) to walk by directly holding and moving its front two legs. The teaching data from all of the instructors is classified into groups using clustering. A representative teaching data set is obtained for each group using averaging. The robot selects an appropriate solution based on some evaluation criteria (the distance moved forward and the impact shock for the body) and then reproduces the movement. This work seems to have similarity to our work in the sense that the student senses the actions of multiple teachers in order to learn a new skill. However, it differs from our method in three respects. First, all of the teachers have the same body morphologies and they differ only in their teaching styles. Second, the student's imitation is not based on a visual observation of a teacher demonstration. Third, the student assumes that its self-model is known *a priori*.

The experimental setup used in all the above approaches comprises a student learning from either a single teacher or a group of morphologically identical teachers. The student robot does not determine whether or not the teacher has an embodiment that is similar to its own embodiment. However, a more complex and a realistic social interaction scenario would consist of a student robot surrounded by multiple, yet morphologically different teacher robots. As a student's ability to learn a new skill from a teacher is affected by how well, or how closely, the student can imitate the teacher's action, it is very important that the student is endowed with an ability to distinguish an appropriate from an inappropriate teacher. Also, in all the above approaches, the student either knows its body structure *a priori* or learns a sensory-motor map to compute its forward kinematics.

Our attempt in this work is to solve some of these limitations of the above approaches. A two-DOF robot crawler is used as a student robot. It is surrounded by two teacher robots whose body plans are identical with and different from that of the student, respectively. The student robot uses a stereo camera rig to observe the actions of a candidate teacher. In the sections that follow, we describe our proposed social robotics framework, comprising

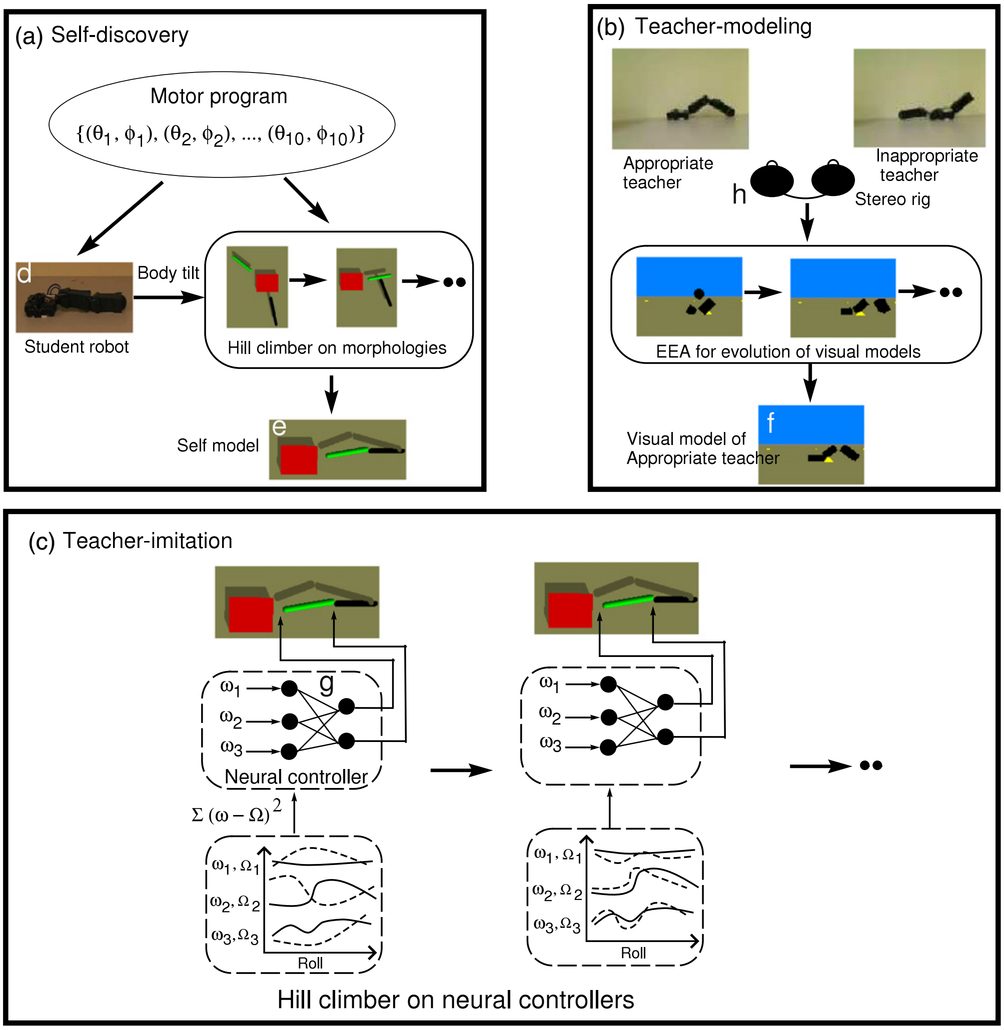

Fig. 1. Framework overview: (a) Self-discovery; the student robot uses actuation-perception relationships to infer its own body structure. (b) Teacher-modeling; the student robot models a candidate teacher. (c) Simulating imitation; the student robot learns a neural controller that allows its self model (evolved in the first phase) to imitate the action of the teacher's visual model (evolved in the second phase). (d) Performing imitation; the student robot uses the learnt neural controller to move, resulting in imitation. (e) Evolved self model of the student. (f) Evolved visual model of the appropriate teacher. (g) Neural controller used to actuate the two joints of the self model. (h) Left and right physical cameras.

the three phases of self-discovery, teacher-modeling, and teacher-imitation, and demonstrate how it is used by the student robot to determine whether or not a candidate teacher is an efficient teacher to learn from.

3. Methods: self discovery, teacher modeling, and teacher imitation

The proposed social-developmental framework that allows a student robot to identify an appropriate teacher from a set of external social agents is outlined in Fig. 1. First, the student performs self-discovery (Fig. 1(a)), a phase in which it uses actuation-perception relationships to infer its own structure. Second, during the phase of teacher modeling (Fig. 1(b)), the student models a candidate teacher using a vision-based, active learning approach to create an approximate physical simulation of the teacher. Third, during the phase of teacher imitation (Fig. 1(c)), the success of finding a neural controller that allows its self model (evolved in the first phase, Fig. 1(e)) to imitate the action of the teachers visual model (evolved in the second phase, Fig. 1(f)) determines whether or not the observed agent is an appropriate teacher. Fourth, the student uses the neural controller (evolved in the third phase) to move, resulting in imitation of the appropriate teacher. Each one of the first three phases is composed of a stochastic optimizer: For self-discovery, the robot optimizes self-models; for teacher modeling, it optimizes visual models of an observed teacher; and for teacher imitation, it optimizes neural controllers used to control its movements.

3.1. Physical robots

The student robot is a two-DOF robot crawler capable of locomotion by exploiting differential friction between its body and foot; it is composed of a main body, an upper limb, and a lower limb. The upper limb is attached to the main body and the lower limb is attached to the upper limb. The two joints are actuated by servo motors that can rotate the joints between −90° and 90°. The default position for the motors (0°) causes the robot to lie flat. Positive and negative angle commands cause each limb to rotate downward and upward, respectively. The robot is equipped with tilt sensors that can measure the orientation of the main body: how much it is tilted to the left or right and how much forward or backward. The body plan for the appropriate teacher (Fig. 2(b)) was chosen to be identical to that of the student. However, the inappropriate teacher has a different body plan (Fig. 2(c)): both its limbs are attached to either side of the main body.

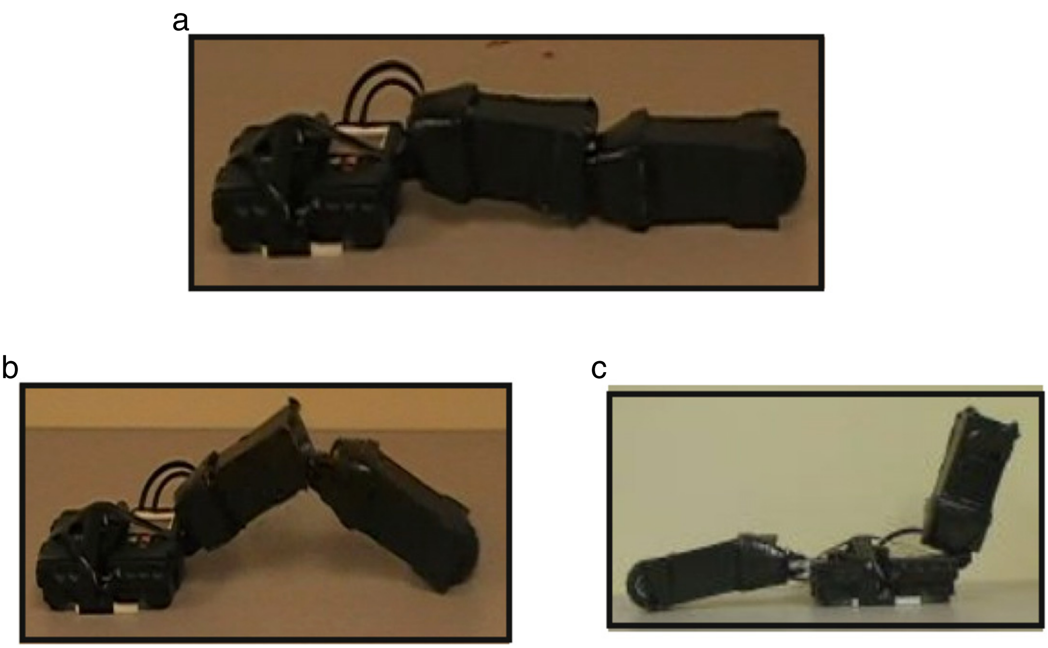

Fig. 2. Physical robots: (a) The student robot. (b) The appropriate teacher. (c) The inappropriate teacher.

3.2. Self discovery

The student knows, *a priori*, that its morphology is composed of three parts that are attached to each other by two actuated rotational joints; it also knows about the physical properties – shape, size, and density – of each part. However, it is ignorant of how the three body parts are attached to each other. The student performs self-discovery using a simplified version of the self discovery method described in Bongard, Zykov, and Lipson (2006): The actuation-perception relationships extracted from the physical robot's response to random actuation commands are used by a simulated student to evolve its self model.

The student robot begins with a set of assumptions about its morphology. Those aspects of its morphology that are unknown (in this case which body parts are connected, and how) form the space of self-models that the student will search through. These assumptions include:

1. the student's morphology is an articulated structure composed of three parts (the main body, limb l_a, and limb l_b);
2. the main body is a rectangular solid whose dimensions are 14 cm × 11 cm × 5 cm;
3. limb l_a is a capped cylinder with radius 1 cm and height 20 cm;
4. limb l_b is a capped cylinder with radius 1 cm and height 20 cm;
5. the density of each part is 0.6 cm^3;
6. each part is attached to one and only one other part by using an actuated rotational joint;
7. limb l_a can attach only to the main body and at an angle $\alpha \in [0°, 360°]$; and
8. limb l_b can attach either to l_a ($\lambda = 0$) or to the main body ($\lambda = 1$) with an angle of attachment in $\beta \in [0°, 360°]$.

3.2.1. Actuation-perception relationships

Initially, the two joints J_a and J_b of the physical student (Fig. 2(a)) are allowed to respond to a set of 10 randomly-generated actuation commands $\{(\theta_j, \phi_j) : j = 1, 2, \ldots, 10\}$, and the resulting orientation of its main body (rx_j, ry_j) for each action command (θ_j, ϕ_j) is recorded. The resulting set of actuation-perception relationships is bundled as the set ASR $= \{(\theta_j, \phi_j, rx_j, ry_j) : j = 1, 2, \ldots, 10\}$. For illustration purpose, the physical student's responses to commands $(-49°, 29°)$, $(-54°, 0°)$, and $(-46°, 48°)$ are shown in Fig. 3(b)–(d), respectively.

3.2.2. Self models

Next, the student robot creates self models of itself in a three-dimensional, physically realistic simulation environment.[2] For each self model constructed, the simulator is populated by the three body parts, and attached together using the parameters stored in a vector (α, λ, β). Note that each instance of a vector maps to a different self model.

3.2.3. Optimizing self models

A hill climber is used to search the space of self models. The set of actuation commands from ASR are fed to the joints of each self model $\text{SM}^{(i)}$. Note that for each $\text{SM}^{(i)}$, the same actuation command (θ_j, ϕ_j) may give rise to a different orientation $(sx_j^{(i)}, sy_j^{(i)})$ of the main body. Therefore, the error $e^{(i)}$ of each self model $(\alpha^{(i)}, \lambda^{(i)}, \beta^{(i)})$, is the offset in the orientation of $\text{SM}^{(i)}$ from that of the physical robot in response to the common actuation commands from ASR:

$$e^{(i)} = \sqrt{\sum_{j=1}^{10} (rx_j - sx_j^{(i)})^2 + (ry_j - sy_j^{(i)})^2}. \tag{1}$$

The search process starts by randomly initializing the genome ($\alpha^{(0)} = 145°, \lambda^{(0)} = 1$, and $\beta^{(0)} = 285°$ for one of the sample runs). The genome is mapped to the self model $\text{SM}^{(0)}$ as shown in Fig. 3(e). Its responses to the sample commands $(-49°, 29°)$, $(-54°, 0°)$, and $(-46°, 48°)$ are shown in Fig. 3(f)–(h), respectively. The genome's error $e^{(0)}$ is evaluated using (1). Next, the genome is subjected to a random mutation and then evaluated. If the child produced by the mutation has an error lower than that of the parent, the parent is replaced by the child; otherwise, the child is discarded. The mutation is achieved by randomly selecting from one to three variables in the genome and perturbing their values using a Gaussian distribution. For the binary integer variable λ, if it is selected for mutation, its value is replaced with a new random value chosen from [0, 1]. The evolved self models at the midpoint

[2] www.ode.org.

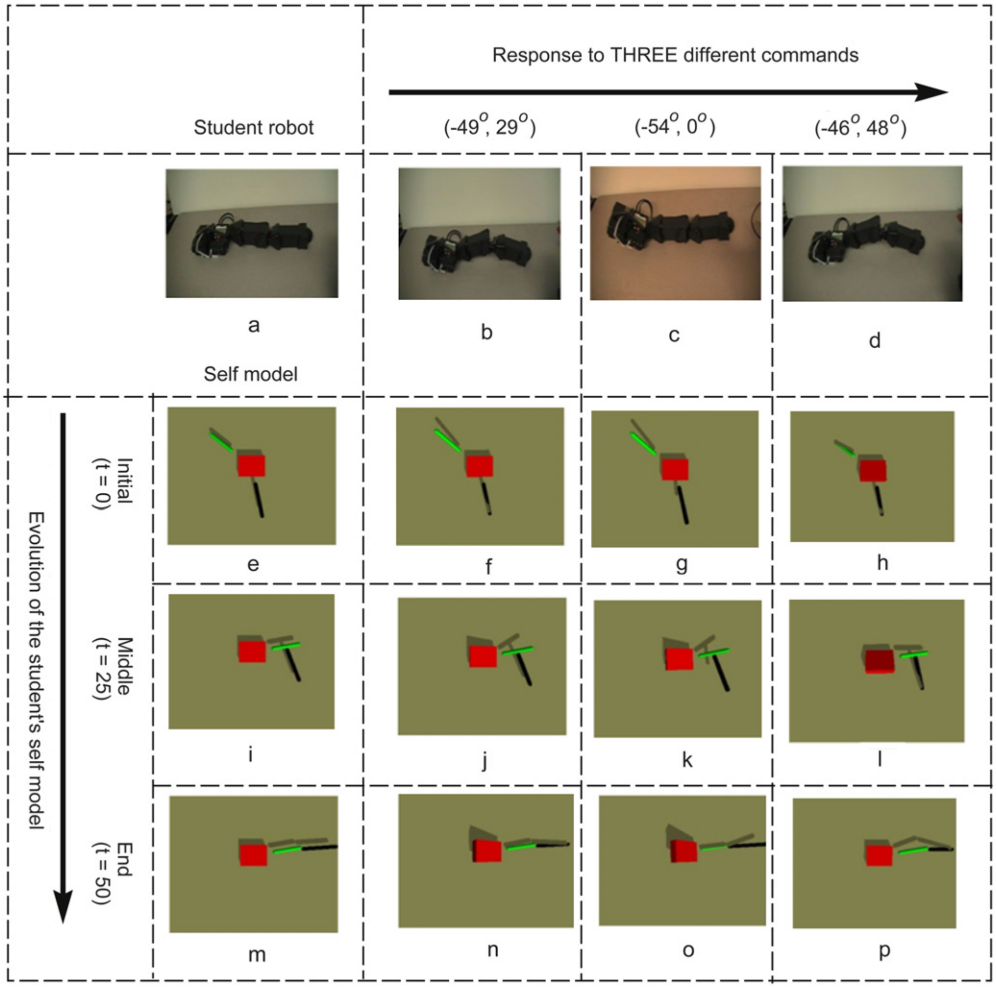

Fig. 3. Self discovery: (a) Physical student robot. (b)–(d) Response of the physical robot to actuation commands (−49°, 29°), (−54°, 0°), and (−46°, 48°), respectively. (e) The randomly initialized self model $SM^{(0)}$. (f)–(h) Response of $SM^{(0)}$ to actuation commands (−49°, 29°), (−54°, 0°), and (−46°, 48°), respectively. (i) The self model $SM^{(25)}$ evolved at the middle of the self discovery phase. (j)–(l) Response of $SM^{(25)}$ to the three actuation commands. (m)–(p) The self model $SM^{(50)}$ evolved at the end of the self discovery phase and its response to the three actuation commands.

$SM^{(25)}$ and at the end $SM^{(50)}$ of the search process are shown in Fig. 3(i) and (m), respectively; the corresponding responses to the three actuation commands are shown in Fig. 3(j)–(l) and (n)–(p), respectively. Note that the evolved self model $SM^{(50)}$ ($\alpha^{(50)} = 6°$, $\lambda^{(50)} = 0$, $\beta^{(50)} = 365°$) of the simulated student and its responses to the action commands closely approximate that of the physical robot. In this way, the student uses self discovery to determine its own body structure.

Equipped now with an autonomously generated self model, the student observes a candidate teacher and attempts to synthesize a physical simulation of it.

3.3. Teacher modeling

We describe here how the student robot uses a stereo vision based active learning approach to observe a teacher robot (Section 3.3.1) and create a 3D physical simulation of it. A rudimentary version of this process was described in Kaipa, Bongard, and Meltzoff (2009). We assume that the student robot is equipped with left and right cameras (Fig. 1(h)) that are trained on the scene,[3] and that we can extract arbitrary pixels from images taken by them. Our approach is based on carefully selecting a small subset of the total pixels available for creating the teacher models (Section 3.3.2). A brief overview of the method of optimizing the teacher models follows (refer to Section 3.3.3 for more details).

We employed the estimation–exploration algorithm (EEA) (Bongard et al., 2006) to create the teacher models. The EEA is a stochastic optimization approach to active learning. One stochastic optimization algorithm improves a population of models to explain a set of training data; a second stochastic optimization method improves a population of unseen training points to induce disagreement among the current model population. Periodically, an optimized training point that induces model disagreement is labeled by the target system being modeled, and included in the training set. This addition deforms the search space in the model population as models must now be optimized to explain both the original data as well as the new training point. In this application, when the student robot models a candidate teacher it maintains two models: each is a three-dimensional simulation of the teacher, similar to those simulations it has created of its own body plan. Fig. 4 depicts several of these teacher simulations.

A pixel with an (x, y) index into the student's camera pair is considered to be a training datum: the grayscale (henceforth referred to as luminosity) indices of a pixel at that position in both cameras is considered to be the two labels associated with that pixel. For each candidate teacher simulation, two simulated cameras are placed in the simulation, and each of the training

[3] The student uses two cameras that are horizontally separated by 10 cm and whose axes are parallel in order to capture video footage of the teacher's act of forward locomotion.

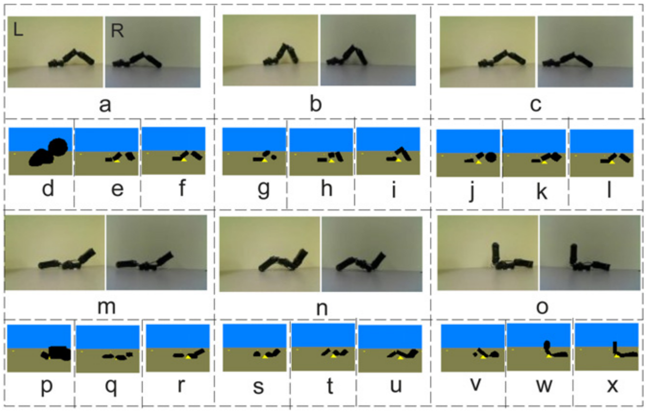

Fig. 4. Visual modeling of an external social agent: (a)–(c) A sequence of three image pairs extracted from the left and right videos of the appropriate teacher's act of forward locomotion. (d) The left image of the randomly initialized model *A*. (e), (f) The left images of model *A* evolved at the middle and end of the first pass. (g)–(i) Second pass of visual model evolution. (j)–(l) Third pass of visual model evolution.

pixel positions is queried for these cameras. The error of a teacher simulation is then the error between the luminosities of the pixels drawn from the student's physical cameras and those drawn from the virtual cameras in the teacher simulation.

3.3.1. The physical teacher

In the experiments described here, two physical teachers were used. The first, referred to as the appropriate teacher, has a mechanical make-up identical to that of the student robot (Fig. 2(b)). The second physical teacher robot, henceforth referred to as the inappropriate teacher, is made of the same three parts as the student and appropriate teacher, but has a different topology: both its limbs are symmetrically attached to either sides of the main body (Fig. 2(c)).

Fig. 4(a)–(c) shows a sequence of image pairs extracted from the two video streams when the student robot observed the appropriate teacher demonstrating a behavior to be imitated.

3.3.2. Teacher models

Each teacher model takes the form of a three-dimensional simulation, into which are placed several 3D shapes that together describe the physical teacher robot. Since the student robot knows that it is composed of three body parts, it attempts to model a candidate teacher with the same number of body parts. If a candidate teacher has a different number of body parts from the student robot, it will not be modeled as well as a candidate teacher with the same number of body parts.

Each model is encoded as a number of parameters; these parameters are used to construct the teacher simulation. In particular, each object of a model is parameterized by that object's shape O_t (a rectangular cuboid, sphere, or capped cylinder), position (x, y, z), orientation (ω (roll), ρ (pitch), μ yaw), and size (d_1 encodes the radius if the object is a sphere, d_1 and d_2 encode the radius and length if the object is a capped cylinder, and d_1, d_2 and d_3 encode the length, width and height if the object is a rectangular cuboid).

In the experiments described here, the pitch and yaw of each object are kept constant ($\rho = \mu = 0$); only the teacher's body parts roll around the axis defined by the line of sight from the cameras to the teacher are modeled by the student. This methodological decision was made to keep the teacher model simple by removing two variables, and it seems likely (although not yet proven) that the teacher's perceived rotation relative to the student's line is more informative than its rotation about other directional vectors.

A model of a candidate teacher with $k = 3$ body parts is thus defined as the vector of parameters

$$\{O_t^{(1)}, x^{(1)}, y^{(1)}, z^{(1)}, \omega^{(1)}, d_1^{(1)}, d_2^{(1)}, d_3^{(1)}, \ldots, \\ O_t^{(k)}, x^{(k)}, y^{(k)}, z^{(k)}, \omega^{(k)}, d_1^{(k)}, d_2^{(k)}, d_3^{(k)}\} \quad (2)$$

where (k) indicates that that parameter describes the kth body part. These parameters are used to construct a 3D compound object in a virtual space.

3.3.3. Optimizing the teacher models

With the above framework in place, the modeling problem can now be solved by carrying out a stochastic search over the space described by the genome. One possible objective function is computed as the root mean squared error of the individual pixel luminosities extracted from the model and those extracted from the target images evaluated over all the pixels. However, the process of requesting luminosities from all of the pixels and using this large set of data in order to compute the above objective function would demand a high computational cost, leading to slow reaction times on the part of the student robot.

In order to overcome this problem, we have here used the estimation–exploration algorithm (EEA) in which several independent models are optimized against a growing set of training pixels, instead of the full pixel array. In short, images are taken by the virtual cameras in the each of the teacher simulations periodically; a pixel is then found that has very different luminosities in each image, and is added to the training set. This is referred to as the exploration phase. Between these pixel extraction events, the teacher simulations are optimized against this growing set of training data using stochastic optimization. This process is referred to as the estimation phase.

The EEA was introduced by Bongard and Lipson (2005) as an application-independent coevolutionary algorithm that automates both model inference and the generation of useful training data. The EEA builds on query by committee (Seung, Opper, & Sompolinsky, 1992), which demonstrated that the optimal method for choosing new training data is through model disagreement. The EEA uses a stochastic optimization process to optimize the model set, and a second optimization process to find potential training data that induces model disagreement. Model optimization and the search for new training data alternates during modeling. More specifically, each pass of the EEA consists of two optimization phases of *exploration* and *estimation*.

In the general case, during the exploration phase candidate training data are sought using stochastic optimization such that, when supplied to the current model set, the outputs of the models are maximally divergent. This ensures that if this disagreement-inducing training data is labeled by the target system, its response will indicate which models have hidden inaccuracies. These inaccurate models will then be automatically replaced during subsequent model optimization in the estimation phase. In contrast to this, in the field of identification for control (Gevers, 2002), training data is passively collected from the target system and the model is not used to determine which experiment to perform next. However, the EEA intelligently chooses new training data so that the amount of data extracted from the target system is minimized, yet sufficiently accurate models of it are produced.

The exploration phase: At the outset of each attempt by the student robot to model a candidate teacher, it first creates two random models of the teacher *A* and *B* that are random instances of the parameter vector shown in Eq. (2). Each of the two models is transformed into 3D simulations, and a left and right virtual camera are placed in each simulation. A sample image taken by the left camera of a random simulation is shown in Fig. 4(d). Images taken by the physical student robot as well as those taken within teacher simulations have a resolution of 100×75 pixels.

A first training pixel is then sought that has different luminosities between the left images taken within simulations A and B, as well as between the right images of A and B. This is accomplished by first creating a candidate pixel set C_p by selecting 1000 pixels using a uniform distribution out of these 7500 pixels. Each pixel from this set is then evaluated to determine how much disagreement it induces in the models in terms of their corresponding luminosities at that pixel's position. For example assume that both models describe a small dark object that appears near the center of the image, but the object's position is slightly different in the image of each model. Then, a pixel chosen near the image center will cause model agreement (both the models will predict that that pixel will be dark in the left image taken from the observed scene), as will a pixel taken near a corner of the image (both the models will predict that that pixel will be light in the left image taken of the observed scene). However, a pixel near the object edges in the models may cause them to disagree (one may predict that pixel will appear dark, while the other model will predict it will appear light). It is these latter pixels that are useful for uncovering hidden inaccuracies in the models.

Based on this observation, the error of a candidate pixel i is computed as

$$\delta_i = |\lfloor(\ell_i^{m_1}/255)\rfloor - \lfloor(\ell_i^{m_2}/255)\rfloor| \tag{3}$$

where $\ell_i^{m_k}$ indicates the luminosity of pixel i in the left image of teacher model k. (This variable ranges in [0, 255].) The pixel among C_p with the highest value for δ therefore induces maximum disagreement across the set of pixels C_p, and is defined herein as the most distinguishing pixel. From (3), note that the pixel luminosities are collapsed into binary values ('1' for white and '0' for all other gray shades) before evaluating the disagreement. Therefore, δ_i is a binary-valued function; that is, the two models either totally agree ($\delta_i = 0$) or totally disagree ($\delta_i = 1$) at pixel i.

The most distinguishing pixel is located for the left images, and the most distinguishing pixel is located for the right images. The corresponding pixels in the left and right images captured by the student robot of the candidate teacher are then queried to extract their luminosities, and are added to the training set. As at each pass of the exploration phase only 1000 out of 7500 possible pixels are considered, only 13.33% of the images are used. Since finding the most distinguishing pixel is the computational bottleneck in this algorithm, this active learning component leads to an almost $10\times$ speed-up in the running time of the algorithm.

Once these pixels are found the estimation phase begins, which optimizes the teacher models for a short period. After this period of model optimization, the exploration phase is again executed, with a new randomly-chosen set of 1000 candidate training pixels.

The estimation phase: In the first pass through the estimation phase, the two random models are trained against the two most distinguishing pixels just discovered during the exploration phase. In subsequent passes through the estimation phase, the two models optimized in the previous pass are re-optimized against the original training pixels as well as the two new pixels obtained during the just-completed exploration phase.

We use a hill climber for teacher model optimization. Let $D_p^l(t)$ and $D_p^r(t)$ be the sets of distinguishing pixels obtained from the left and right images of candidate teachers after t passes through the exploration phase, respectively. The error of a candidate teacher model can then be defined as below:

$$L_{\text{rmse}}(t) = \frac{\sqrt{\sum_{i=1}^{N_{dp}^l(t)} (\ell_i^{l_t} - \ell_i^{l_m})^2}}{N_{dp}^l(t)} + \frac{\sqrt{\sum_{j=1}^{N_{dp}^r(t)} (\ell_j^{r_t} - \ell_j^{r_m})^2}}{N_{dp}^r(t)} \tag{4}$$

where $i \in D_p^l(t), j \in D_p^r(t)$, and $N_{dp}^l(t), N_{dp}^r(t)$ are the number of distinguishing pixels collected up to and including iteration t corresponding to the left and right images, respectively.

At each iteration, the genome of each model is subjected to random mutation and then evaluated. If the child model produced by the mutation has a lower error than the parent model, the parent is replaced with the child; otherwise, the child is discarded. The mutation is achieved by randomly selecting from one to five variables in the genome and perturbing their values using a Gaussian distribution. Since the variables are randomly selected, a variable may undergo mutation more than once during any iteration. For the integer object shape variable, if it is selected for mutation, its value is replaced with a new random value chosen from [0, 1, 2]. If an object changes shape due to mutation, the size parameters are altered so that the visual impact on the new image is slight, and there is therefore little discontinuity between successive model images. For instance, if a sphere is switched to a cuboid, the length, height and width of the new cuboid are set to the original sphere's circumference. The above optimization scheme steps through $N_{\text{est}} = 8$ iterations.

Estimation–exploration algorithm flow: For each run in which the student robot attempts to model a candidate teacher, it alternates between the exploration and estimation phase 1000×3 times; one execution of the exploration phase followed by one execution of the estimation phase is defined here as one pass through the run. In the first pass, the teacher models are optimized against two pixels (one from the left and right images respectively (Fig. 4(a))). Over the next 1000 passes, a training buffer that can hold 2×1000 pixels is gradually filled.

Once the training buffer is filled, the student takes a new pair of images of the candidate teacher (Fig. 4(b)), and discards the original two images used to assign luminosities to the first 2×1000 training pixels. Over the next 1000 passes, the oldest pixel pair in the training buffer is overwritten by a new pixel pair produced by the estimation phase and assigned luminosities by the new image pair. Once 1000 passes have elapsed, a new image pair (Fig. 4(c)) is obtained by the student robot and used for assigning luminosities to training pixels. This process continues until 3000 passes in total have elapsed. Images are periodically refreshed by the student robot in this manner because it is assumed that the candidate teacher is moving.

Over the course of optimizing teacher models (Fig. 4(d)–(l) and (p)–(x) for appropriate and inappropriate teachers, respectively), the evolutionary changes across the 1000 passes in the best of the two model's parameter sets are recorded: systematic changes in parameters over the modeling process reflect indirectly the motion of the candidate teacher, and is used in the third phase: teacher imitation.

3.4. Teacher imitation

In this section the way in which the student uses the results from its self-discovery and teacher-modeling phases to imitate the observed teacher is described. The student's ability to learn a new skill from a teacher requires first that the student is physically able to imitate a candidate teacher. Therefore, it is imperative for the student to identify appropriate teachers and learn to ignore any inappropriate teachers in its surroundings.

Consequently, in this phase (Fig. 1(c)), the student attempts to optimize a neural controller (Section 3.4.1) such that its self model (evolved in the first phase) imitates the actions of the teacher's model (evolved in the second phase). If the student's self-model eventually succeeds in imitating the teacher's actions, then it concludes that the observed agent is an appropriate teacher; otherwise, it concludes that the agent is an inappropriate teacher.

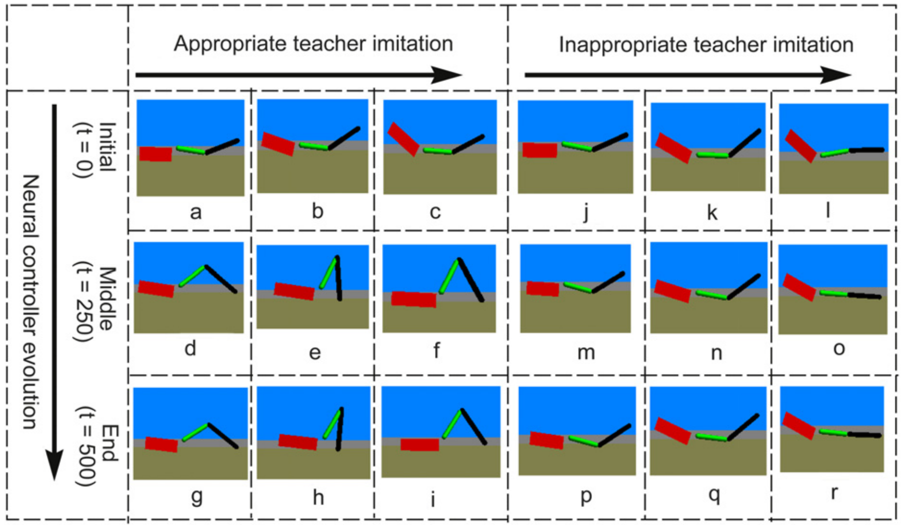

Fig. 5. Imitation of a teacher: (a)–(c) Self model's attempt to imitate the appropriate teacher's action by using the randomly initialized neural controller. (d)–(f) Its imitation attempt by using the neural controller evolved at the middle of the simulation run. (g)–(i) The self model succeeds in evolving the neural controller that allows itself to closely imitate the appropriate teacher. (j)–(r) Self model's unsuccessful attempt to imitate the inappropriate teacher's action at the beginning, middle, and end of the neural controller evolution.

3.4.1. *Neural controller*

The student uses a simple 3×2 single layer neural network (see Fig. 1(j)) to control the actions of its self model. The roll histories (roll is defined here as the rotation about an axis that is parallel with the student's line of sight; roll history of an object refers to the changes in its roll over the modeling process) of the three objects that constitute the evolved visual model of the observed teacher are fed to the three inputs of the neural network. Each roll history is sub-sampled at every 100 intervals to obtain an input length of 240. (The total length of each parameter history in the recorded model parameter set can be computed as $8(N_{est}) \times 1000 \times 3 = 24\,000$.) The two outputs of the neural network are used to actuate the two rotary joints of the self model evolved in the first phase; the outputs are passed through an exponential activation function and linearly scaled to generate angle commands in the range $[-90^\circ, 90^\circ]$.

3.4.2. *Optimizing the neural controller*

The student robot faces two challenges when attempting to imitate a candidate teacher: its body parts must rotate in the same manner as the candidate teacher, but it is not sure which of the teacher's body parts correspond to its own body parts. The following approach resolves both of these issues.

An initial random neural network is created using a randomly-generated parameter vector $\{w_{11}, w_{12}, w_{13}, w_{21}, w_{22}, w_{23}\}$ which encodes each weight in the network. Each weight is randomly initialized with a value between −1 and 1. First, the best self model from the first phase is created in the simulator. Second, for each time step t, the neural network is provided at its input layer with the roll triplet $(\omega_1^{(t)}, \omega_2^{(t)}, \omega_3^{(t)})$, where $\omega_i^{(t)}$ indicates the roll of the teacher's ith body part recorded at tth position in the teacher's roll history. Third, the values are propagated from the input to the output layer. Fourth, the two output values are fed to the joints of the student's self model. Fifth, the resulting roll response of the three body parts of the self model $(\Omega_1(t), \Omega_2(t), \Omega_3(t))$ at this tth time step of the simulation are recorded. Finally, after 240 time steps of simulation, the quality of the neural network is computed.

The error of the neural network is evaluated based on how well the self model's movements approximate that of the teacher's visual model. In particular, the error is computed as the error between the roll of each body part of the self model and that of its corresponding object in the teacher's visual model. This object correspondence is first resolved in order to decide which body part of the self model should be matched with which object of the visual model.

For this purpose, the self model is aligned such that the main body is placed to the extreme left of its visual field and body part l_b is placed to the extreme right. Now, the Euclidean distances between the normalized positions of the body parts of the self model and the normalized positions of the three objects from the teacher simulation are computed in order to determine the object correspondences based on their proximity to each other. Let (ν_1, ν_2, ν_3) be the new roll triplet of the teacher's visual model after reordering. For example, if the computed object order is (2, 1, 3), then $(\nu_1 = \omega_2, \nu_2 = \omega_1, \nu_3 = \omega_3)$. Now, the error of a neural network k is given by:

$$e_k = \sqrt{\sum_{i=1}^{240} (\nu_1^i - \Omega_1^i)^2 + (\nu_2^i - \Omega_2^i)^2 + (\nu_3^i - \Omega_3^i)^2}. \quad (5)$$

A hill climber is used to optimize the weights of the neural network using e_k as an objective function. At each iteration, the set of weights is subjected to random mutation and then evaluated as described above. If the new neural network achieves a lower error than the parent network, the parent is replaced with the child; otherwise, the child is discarded. Mutation is achieved by randomly selecting one to three weights and perturbing their values using a Gaussian distribution. Optimization is terminated after 500 iterations.

Fig. 5 reports results from a typical attempt by the student robot to imitate both the appropriate and inappropriate teachers. Fig. 5(a)–(c) shows the self model's attempt to imitate the appropriate teacher's action using the initial, randomly-initialized neural controller. Note here that the self model is unable to replicate the teacher's actions. However, Fig. 5(d)–(f), which reports the self model's actions after 250 iterations, and Fig. 5(g)–(i), which reports the self model's actions after 500 iterations, indicate that the self model quickly learns to closely imitate the movements of the observed teacher.

Fig. 5(j)–(r) shows the self model's response when attempting to imitate a inappropriate teacher. Note that the self model fails to find a neural controller that allows itself to imitate this candidate teacher.

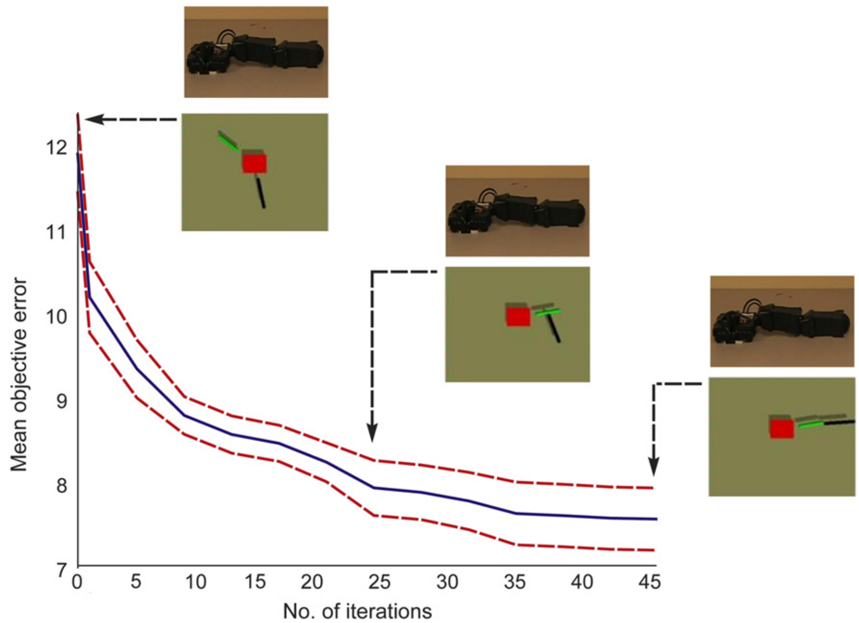

Fig. 6. Objective error of evolved self models of the student averaged over 30 independent trials. (The standard error of the mean graphs are shown above as dotted curves.) The self models at the beginning ($t = 0$), middle ($t = 25$), and end ($t = 50$) of a sample evolution are shown above for illustration purpose.

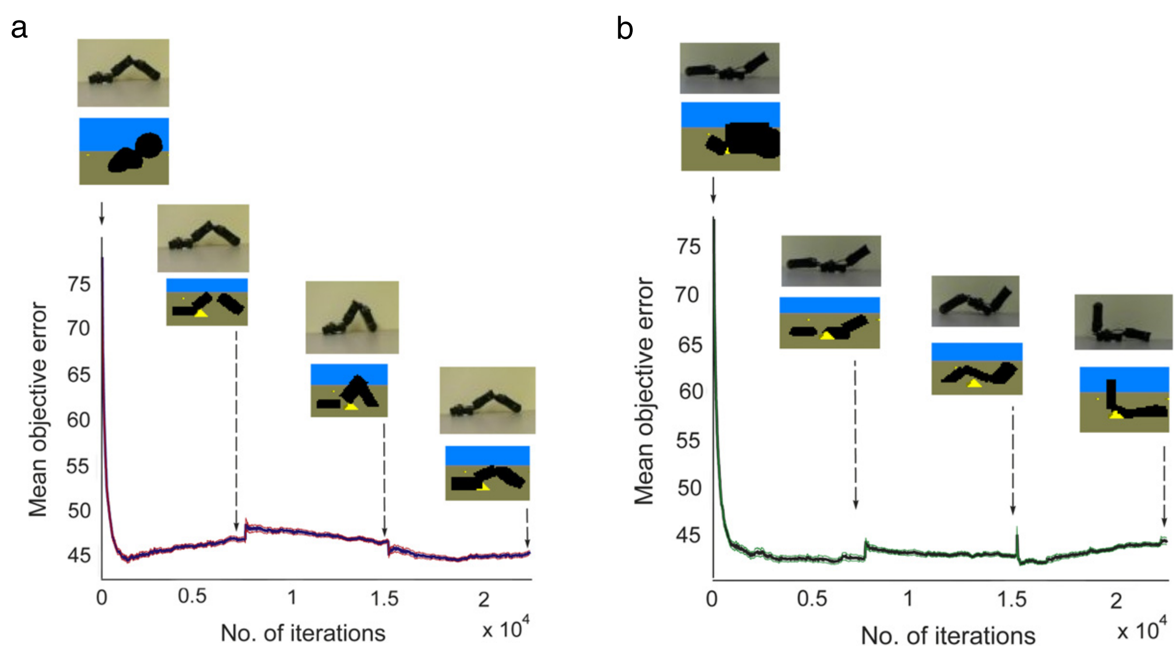

Fig. 7. Objective error of evolved teacher models averaged over 30 independent trials: (a) The appropriate teacher. (b) The inappropriate teacher. (The standard error of the mean graphs are shown above as dotted curves. The visual models at the beginning, and at the end of each pass of a sample evolution are shown above for illustration purposes.)

4. Results

In the previous section we have used results from a single typical experiment to demonstrate how each of the three phases – self-discovery (Fig. 3), teacher modeling (Fig. 4), and teacher imitation (Fig. 5) – to together enable a student robot to distinguish between appropriate and inappropriate teachers. We conducted thirty independent trials of each of the three phases in order to evaluate the statistical consistency of our method.

Fig. 6 shows the objective error of evolved self models of the student averaged over 30 independent trials. We define objective error as the actual errors inherent in the self model compared to the physical robot, and distinguish it from subjective error, which is the error inferred by the student robot given the sensor data made available to it (Eq. (1)). The standard errors of the mean (SEM) graphs are shown as dotted curves. The self models at the beginning ($t = 0$), middle ($t = 25$), and end ($t = 50$) of a sample run, shown in the figure, illustrate the reduction in objective error: that is, how well the morphologies of the self models starts to approximate the morphology of the physical student. The statistically significant difference between the mean objective error at the beginning ($t = 0$) and the termination ($t = 50$) of the self-modeling phase indicates that the student robot is consistently able to create a simulation of its own body.

Fig. 7(a) and (b) report the objective error of the models of the appropriate teacher and the inappropriate teacher, respectively,

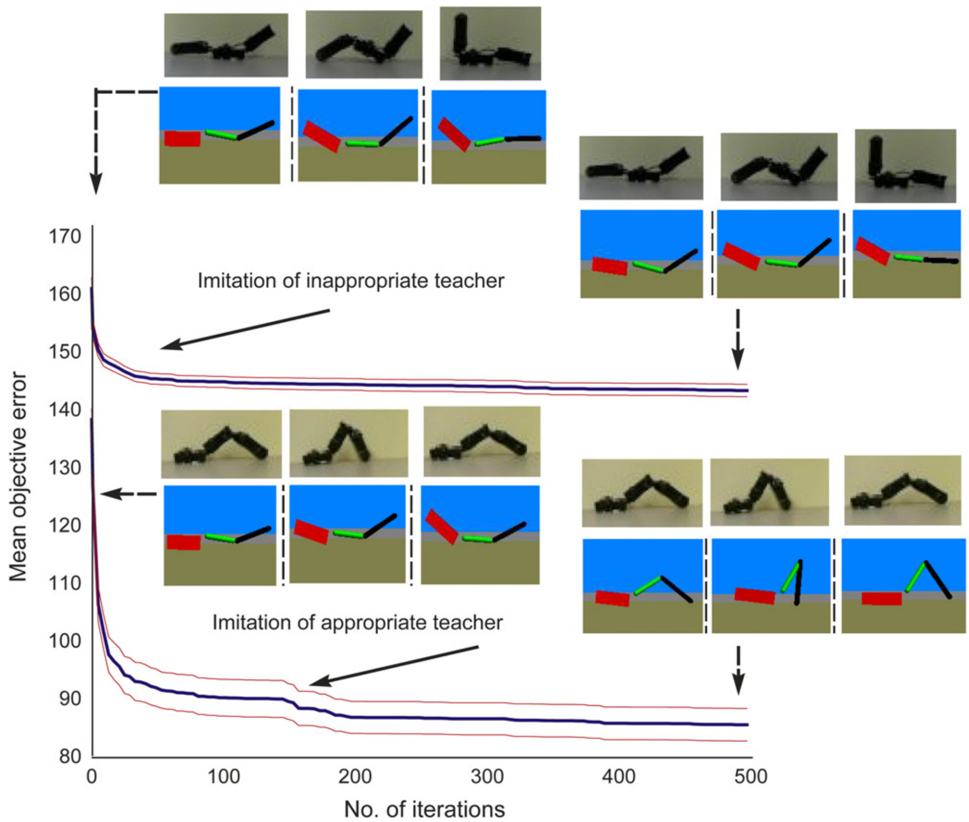

Fig. 8. Objective error of evolved neural controllers for the appropriate teacher and the inappropriate teacher averaged over 30 independent trials. (The standard error of the mean graphs are shown above as dotted curves.) The self model's imitation attempt at the beginning and the end of a sample neural controller evolution are shown above for both the teachers.

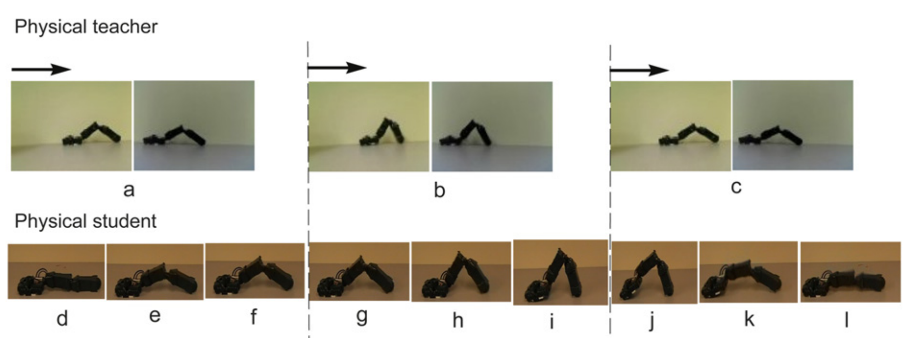

Fig. 9. (a)–(c) Input to the teacher modeling module: Stereo image pairs extracted from the left and right videos of the appropriate teacher's act of forward locomotion. (d)–(l) The physical student uses the neural controller (evolved in the third phase) to move, resulting in the successful imitation of the appropriate teacher.

averaged over 30 independent trials. Several teacher models obtained at different points during the optimization process are added for comparison purposes. Again, the significant difference between the mean objective error of the teacher models at the outset and at the termination of this phase, for both candidate teachers, indicates that the student robot consistently converges on an approximate model of the candidate teacher.

Fig. 8 indicates the mean objective errors of optimized neural controllers for the appropriate teacher and the inappropriate teacher, averaged over 30 independent trials. The self model's imitation attempt at the beginning and the end of a sample neural controller evolution are shown for both the teachers. Note that the mean objective error corresponding to the inappropriate teacher is almost flat and there is no statistically significant difference between the initial and final mean objective errors. However, there is a significant difference in the case of the appropriate teacher. This demonstrates that the student is consistent in distinguishing the appropriate teacher from the inappropriate one.

Finally, if the student robot is able to significantly reduce the error of a neural network, it may elect to execute that controller using its physical, rather than modeled self. In many cases, this results in imitation of the appropriate teacher. Snapshots from the video footage of a sample experiment in which the student learns to imitate the forward locomotion of the appropriate teacher are shown in Fig. 9(d)–(l).

5. Discussion

We have here demonstrated that a simple physical robot can consistently model itself and other candidate teachers in its environment. It can use these self and other models to determine whether a candidate teacher is an appropriate teacher: that is, whether the student can imitate the actions of the teacher. This was demonstrated using a two degree-of-freedom student robot, an appropriate teacher with the same number of DOFs and body topology, and an inappropriate teacher with the same DOFs but a different body topology.

This approach was designed such that teachers may be deemed appropriate or not even if their visual appearance differs from that of the student robot. For instance the student robot would still be able to imitate teachers of different size and at different distances from itself, as well as different positions in its field of view, because change in the interrelations of the teacher's body parts are used for imitation, rather than absolute geometric properties.

This ability to see beyond the immediate geometric similarities between the self and another opens the possibility for inference of behavior implied by abstract motion by a teacher, such as hand gestures: walking two fingers across a surface could indicate a signal by a teacher to a student to perform bipedal walking. Prior work on gesture imitation (e.g., waving a hand) has shown that limitations on the perception and the execution in a robotic set-up have a profound influence on the number and type of gestures that can be used to socially train robots (Jansen, 2003). However, the teacher modeling phase demonstrated here could be extended to inferring behavior from hand or arm gestures. This has practical applications as a way of 'teaching' a robot by a human user quickly.

Although both the student and teacher robots had simple morphologies in this work, the proposed framework could be scaled up to robots having more complex morphologies. Indeed it was demonstrated by Bongard et al. (2006) that an estimation–exploration algorithm could be employed by an eight DOF articulated robot to self model itself and use these models to recover from body damage. The method demonstrated here could be extended in still other ways. In the teacher modeling phase, the error of a model was taken to be the difference between individual pixels in images taken by the physical student robot of the physical teacher robot, and by virtual cameras in the teacher models. However, other kinds of visual primitives like bounded pixel patches and texture patches could also be used, to make the recognition and modeling of more complex physical objects possible. Furthermore, in the current work only cuboids, spheres, and capped cylinders were used as the basic elements. However, compound objects that are successfully modeled could be added to this basic set of objects so that the algorithm could look for such templates in more complex scenes.

Inferring not just the roll of the teacher's body parts around the student robot's line of sight, but other rotations of the teacher, as well as other changes over time such as its motion, and motion in response to the student, could be incorporated into the modeling process. This latter property of a teacher could provide a social contingency cue to the student: in other words, the student could begin to model under what conditions the teacher responds to the student's actions. Finally, the simplest possible stochastic optimizers – hill climbers – were employed in this work; these could be replaced in future work with more sophisticated optimization methods to speed up modeling and imitation.

6. Conclusions

Solutions to the problem of who the robot should learn from are crucial to the progress of social robotics: a student's ability to learn a new skill from a teacher is affected by how well, or how closely, the student can imitate the teacher's action. We have here proposed and demonstrated a framework that provides one solution to this problem. Specifically, we have shown how a robot can use self-discovery as a means of recognizing self-other equivalences with other robots in order to identify appropriate teachers and avoid wasting time attempting to learn from any inappropriate teachers in its surroundings. Our approach has been tested using real robot experiments in which a physical student robot, not preprogrammed to locomote, observed a teacher locomoting, and learned to move in the same way using the proposed framework.

Acknowledgement

This work was supported by funding from NSF SBE-0751385 and SBE-0354453.

References

Alissandrakis, A., Nehaniv, C. L., & Dautenhahn, K. (2007). Correspondence mapping induced state and action metrics for robotic imitation. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, *37*(2), 299–307.

Amit, R., & Mataric, M. (2002). Learning movement sequences from demonstration. In *Proceedings of int. conf. on development and learning (ICDL-2002)* (pp. 203–208).

Billard, A., & Schaal, S. (2001). Robust learning of arm trajectories through human demonstration. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems* (pp. 734–739).

Bongard, J. C., & Lipson, H. (2005). Nonlinear system identification using coevolution of models and tests. *IEEE Transactions on Evolutionary Computation*, *9*(4), 361–384.

Bongard, J. C., Zykov, V., & Lipson, H. (2006). Resilient machines through continuous self-modeling. *Science*, *314*, 1118–1121.

Breazeal, C., Berlin, M., Brooks, A., Gray, J., & Thomaz, A. L. (2006). Using perspective taking to learn from ambiguous demonstrations. *Robotics and Autonomous Systems*, *54*, 385–393.

Breazeal, C., & Scassellati, B. (2002). Robots that imitate humans. *Trends in Cognitive Sciences*, *6*(11), 481–487.

Butterworth, G. (1992). Origins of self-perception in infancy. *Psychological Inquiry*, *3*(2), 103–111.

Coates, A., Abbeel, P., & Ng, A. Y. (2008). Learning for control from multiple demonstrations. In *Proceedings of the 25th international conference on machine learning* (pp. 144–151).

Cooley, C. H. (1902). *Human nature and the social order*. New York: Charles Scribners.

Dautenhahn, K., & Nehaniv, C. L. (2002). The agent-based perspective on imitation. In K. Dautenhahn, & C. L. Nehaniv (Eds.), *Imitation in animals and artifacts* (pp. 1–40). Cambridge, MA: MIT Press.

Gevers, M. (2002). A decade of progress in iterative control design: from theory to practice. *Journal of Process Control*, *12*(4), 519–531.

Jansen, B. (2003). An imitation game for emerging action categories. In *Proceedings of advances in artificial life, 7th European conference on artificial life* (pp. 800–809).

Jansen, B., & Belpaeme, T. (2006). A computational model of intention reading in imitation. *Robotics and Autonomous Systems*, *54*, 394–402.

Kaipa, K. N., Bongard, J. C., & Meltzoff, A. N. (2009). Combined structure and motion extraction from visual data using evolutionary active learning. In *Proceedings of the genetic and evolutionary computation conference* (pp. 121–128).

Lopes, M., & Santos-Victor, J. (2007). A developmental roadmap for learning by imitation in robots. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, *37*(2), 308–321.

Mead, G. H. (1934). *Mind self and society*. Chicago: Chicago University Press.

Meltzoff, A. N. (2007a). 'Like me': a foundation for social cognition. *Developmental Science*, *10*(1), 126–134.

Meltzoff, A. N. (2007b). The 'like me' framework for recognizing and becoming an intentional agent. *Acta Psychologica*, *124*(1), 26–43.

Meltzoff, A. N., Kuhl, P. K., Movellan, J., & Sejnowski, T. J. (2009). Foundations for a new science of learning. *Science*, *325*, 284–288.

Meltzoff, A. N., & Moore, M. K. (1997). Explaining facial imitation: a theoretical model. *Early Development and Parenting*, *6*, 179–192.

Nyuwa, T., Katagami, D., & Nitta, K. (2008). Robotic social imitation depends on self-embodiment and self-evaluation by direct teaching under multiple instructors. In *Proceedings of IEEE international conference on fuzzy systems* (pp. 2046–2051).

Schaal, S. (1997). Learning from demonstration. In *Advances in neural information processing systems* (pp. 1040–1046).

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Proceedings of the fifth workshop on computational learning theory* (pp. 287–294).

Shon, A., Storz, J. J., Meltzoff, A. N., & Rao, R. P. N. (2007). A cognitive model of imitative development in humans and machines. *International Journal of Humanoid Robotics*, *4*(2), 387–406.