Interactive Robot Learning

-

RSS 2008 WORKSHOP

June 28, 2008, Zürich, Switzerland.

Organizers

Andrea Lockerd Thomaz
Interactive Computing
Georgia Institute of Technology
Atlanta, USA

Henrik Jacobsson
German Research Center for Artificial Intelligence (DFKI GmbH)
Language Technology Lab
Saarbrücken, Germany

Geert-Jan M. Kruijff
German Research Center for Artificial Intelligence (DFKI GmbH)
Language Technology Lab
Saarbrücken, Germany

Danijel Skocaj
Visual Cognitive Systems Laboratory
Faculty of Computer and Information Science
University of Ljubljana
Ljubjana, Slovenia

Preliminary Program

9:00	Intro, Welcome
9:20	Keynote: Jeff Orkin, Cognitive machines group, MIT Media Labs, Cambridge, MA, USA Title: "Learning Models of Social Behavior and Dialogue with The Restaurant Game"
10:00	Q&A,
10:20	coffee break
10:45	S. Calinon & A. Billard, "Robot Programming by Demonstration Framework Integrating Statistical and Social Cues"
11:05	Maren Bennewitz, Tobias Axenbeck, Sven Behnke, & Wolfram Burgard , "Robust Recognition of Complex Gestures for Natural HRI"
11:25	Matthieu Lagarde , Pierre Andry , Philippe Gaussier, & Giovannangeli Christophe, "Learning new behaviors: Toward a Control architecture merging spatial and temporal modalities"
11:45	O. Booij, B. Krose, J. Peltason, T. Spexard, & M. Hanheide "Moving from augmented to interactive mapping"
12:10	lunch break
13:30	Keynote: Jan Peters, Max-Planck Institute of Biological Cybernetics, Tübingen, Germany. Title: "Towards Motor Skill Learning in Robotics"
14:10	Q&A
14:20	Adriana Tapus & Maja Matari, "Active Learning for Socially Assitive Robotics for Stroke Rehabilitation and Dementia Care"
14:40	Ayanna M. Howard, Sekou Remy, & Hae Won Park, "Learning of Arm Exercise Behaviors: Assistive Therapy based on Therapist-Patient Observations"
15:00	Jure Zabkar, Ivan Bratko, Ashok Mohan & Erwin Prassler , "Learning qualitative models by an autonomous robot"
15:20	coffee break
15:45	Keynote: Aude Billard, Learning Algorithms and Systems Laboratory, Ecole Polytechnique Federale de Lausanne, Switzerland Title: "Adaptive Control and Imitation Learning in Robots"
16:25	Q&A
16:35	Panel/open discussion
17:00	End of workshop

1 Welcome

This workshop on Interactive Robot Learning will span the breadth of research questions at the intersection of Machine Learning and Human-Robot Interaction.

Many future applications for autonomous robots bring them into human environments as helpful assistants to untrained users in homes, offices, hospitals, and more. These applica- tions will often require robots to flexibly adapt to the dynamic needs of human users. Rather than being pre-programmed at the factory with a fixed repertoire of skills, these personal robots will need to be able to quickly learn how to perform new tasks and skills from natural human instruction. Moreover, it is our belief that people should not have to learn a new form of interaction in order to teach these machines, that the robots should be able to take advantage of communication channels that are natural and intuitive for the human partner.

Topics:

- Human-Robot Interaction
- Machine Learning
- Learning by demonstration
- Learning by imitation
- Reinforcement learning with human input
- Active Learning
- Communication of knowledge and metaknowledge
- Identification of new requirements for ML in social domains
- Identification of suitable metrics for interactive learning
- User studies on interactive robot learning

2 Invited speakers

2.1 Jan Peters

2.1.1 Talk: Towards Motor Skill Learning in Robotics

Autonomous robots that can assist humans in situations of daily life have been a long standing vision of robotics, artificial intelligence, and cognitive sciences. A first step towards this goal is to create robots that can learn tasks triggered by environmental context or higher level instruction. However, learning techniques have yet to live up to this promise as only few methods manage to scale to high-dimensional manipulator or humanoid robots. In this talk, we investigate a general framework suitable for learning motor skills in robotics which is based on the principles behind many analytical robotics approaches. It involves generating a representation of motor skills by parameterized motor primitive policies acting as building blocks of movement generation, and a learned task execution module that transforms these movements into motor commands.

Learning parameterized motor primitives usually requires reward-related self-improvement, i.e., reinforcement learning. We propose a new, task-appropriate architecture, the Natural Actor-Critic. This algorithm is based on natural policy gradient updates for the actor while the critic estimates the natural policy gradient. Empirical evaluations illustrate the effectiveness and applicability to learning control on an anthropomorphic robot arm.

For the proper execution of motion, we need to learn how to realize the behavior prescribed by the motor primitives in their respective task space through the generation of motor commands. This transformation corresponds to solving the classical problem of operational space control through machine learning techniques. Such robot control problems can be reformulated as immediate reward reinforcement learning problems. We derive an EM-based reinforcement learning algorithm which reduces the problem of learning with immediate rewards to a reward-weighted regression problem. The resulting algorithm learns smoothly without dangerous jumps in solution space, and works well in application to complex high degree-of-freedom robots.

2.1.2 Bio

Jan Peters is a Senior Research Scientist at the Max-Planck Institute for Biological Cybernetics and head of the new Robot Learning Lab (RoLL) in the Schoelkopf Department. Before joining MPI, he received a Ph.D. from the University of Southern California, working at the Computational Learning and Motor Control lab with Stefan Schaal, Sethu Vijyakumar and Firdaus Udwadia. He received a M.Sc. in Computer Science and M.Sc. in Mechanical Engineering from University of Southern California as well as a Diplom-Informatiker from Hagen University and a Diplom-Ingenieur in Electrical Engineering from Munich University of Technology (TU Muenchen). He has been a visiting researcher at Advanced Telecommunication Research Center (ATR), Kyoto, Japan in 2000 and 2003, a visiting researcher at National University of Singapore (NUS) in 2001 and worked as graduate research assistant at the Institute of Robotics and Mechatronics of the German Aerospace Research Institute (DLR) in Oberpfaffenhofen, Germany form 1997-2000. His research interests include robotics, nonlinear control, machine learning, and motor skill learning.

MORE INFO: `http://www.jan-peters.net`

2.2 Aude Billard

2.2.1 Talk: Adaptive Control and Imitation Learning in Robots

A key issue in robot imitation learning is to find "what to imitate", i.e. to determine the key components of a task that are relevant for its completion.

Such information is crucial for a proper generalization over a set of examples. Moreover, it provides a way to speed up learning by reducing the dimensionality of the features' space. However, this issue has received scant attention so far.

In this talk, a framework for extracting the relevant components of a task is presented. It is based on Gaussian Mixture Models (GMM) of multi-dimensional signals in concurrent frames of reference. The relative importance of each part of the signals is estimated through the covariance matrices of the GMM. Gaussian Mixture Regression is then applied to infer an optimal generalized signal which can further drive the reproduction of the task. An extension of this framework for learning a dynamical model of the task as a second order derivative of the end-effector's motion is also presented.

A second crucial component to robot imitation learning is the problem of "how to imitate". On one hand, the robot must find a way to translate the motions demonstrated by the human in motions feasible for its body, whose size and range of motions differ from those of the demonstrator. On the other hand, the robot must be able to adapt its movements to achieve a proper completion of the task when the context differ from the one used during the demonstrations.

In the framework presented here, the control of the robot's motions is provided by a stable dynamical system, active in a hybrid cartesian-joint angle frame of reference. The dynamic nature of the controller ensures on-line determination of the trajectory if perturbations occur and stable convergence to the target. The redundancy of the representation of the motion offers an elegant solution to the joint limit avoidance problem. Reproduction of the task is obtained by modulating the hybrid dynamical systems using the trajectories inferred from the demonstration.

2.2.2 Bio

Aude Billard is Associate Professor and head of the LASA Laboratory at the School of Engineering at the Swiss Federal Institute of Technology in Lausanne. She received her B.Sc. (1994) and M.Sc. (1995) in Physics from EPFL, with specialization in Particle Physics at the European Center for Nuclear Research (CERN), a MSc. in Knowledge-based Systems (1996) and a Ph.D. in Artificial Intelligence from the Department of Artificial Intelligence at the University of Edinburgh. She worked as a Post-doctoral Fellow at IDSIA and LAMI (EPFL, 1998-1999), then as research associate (1999-2000), Research Assistant Professor (2000-2002) at the department of Computer Sciences at the University of Southern California, prior to joining the EPFL.

Learning Algorithm and Systems Laboratory, School of Engineering, EPFL - Swiss Federal Institute of Technology in Lausanne, Lausanne 1015, Switzerland `http://lasa.epfl.ch` Email: `aude.billard@epfl.ch`

2.3 Jeff Orkin

2.3.1 Talk: Learning Models of Social Behavior and Dialogue with The Restaurant Game

We look forward to a future where robots collaborate with humans in the home and workplace, and virtual agents collaborate with humans in games and training simulations. A representation of common ground for everyday scenarios is essential for these agents if they are to be effective collaborators and communicators. Effective collaborators can infer a partner's goals and predict future actions. Effective communicators can infer the meaning of utterances based on semantic context. This talk describes a multiplayer video game used to collect data from thousands of people about everyday scenarios, an unsupervised system that learns statistical models of language and interaction, and first steps towards generating dialogue and behavior from these models. Specifically, the talk will describe learning the restaurant scenario from data collected from over 10,000 players of an online game called The Restaurant Game (`http://theRestaurantGame.net`).

2.3.2 Bio

Jeff Orkin is a PhD student in Professor Deb Roy's Cognitive Machines Group at the MIT Media Lab. Jeff's research focuses on Artificial Intelligence for characters that learn to communicate and collaborate by observing humans playing online multiplayer games. Prior to enrolling at the Media Lab, Jeff developed several generations of A.I. systems in the game industry. As a Senior Engineer at Monolith Productions, Jeff focused on goal-oriented autonomous character behavior and planning, while developing A.I. systems for the award winning titles No One Lives Forever 2 and F.E.A.R. Jeff is a Contributing Author and Section Editor of the A.I. Game Programming Wisdom book series, has presented at the Game Developer's Conference and AIIDE, and holds a Master's degree in Computer Science from the University of Washington and Bachelor's degree in Computer Science from Tufts University with a minor in Studio Art.

A robot programming by demonstration framework integrating statistical and social cues

Sylvain Calinon and Aude Billard

Abstract— **We present a probabilistic approach in robot programming by demonstration that allows to extract incrementally the constraints of a task in a continuous form and to reproduce a generalization of the learned skill in new situations. Throughout this work, we highlight the importance of including the user's teaching abilities in the machine learning process by using different modalities to convey the demonstrations (observational learning and kinesthetic teaching), and by designing human-robot interactive scenarios mimicking the human process of teaching. We then present our current research towards a socially driven statistical learning framework to reduce the complexity of the skill transfer process.**

I. Robot programming by demonstration

Robot Programming by Demonstration (RPD) covers methods by which a robot learns new skills through human guidance. Our research aims at bringing such user-friendly human-robot teaching systems that would speed up the skill transfer process. We present a generic probabilistic framework gathering information from cross-situational observations of a skill with information extracted from different social cues observed during the interaction.

Generic approaches to transfer new skills to a robot are those that allow the robot to extract automatically what are the important features characterizing each task and to search for a controller that optimizes the reproduction of these characteristic features. A key concept at the bottom of these approaches is that of determining a *metric of imitation performance*. A metric of imitation provides a way of expressing quantitatively the user's intentions during the demonstrations and to evaluate the robot's faithfulness at reproducing those. To learn the metric (i.e. infer the task constraints), one common approach consists of creating a model of the skill based on several demonstrations performed in slightly different conditions (cross-situational statistical learning). This generalization process consists of exploiting the variability inherent to the various demonstrations to extract which are the essential components of the task. These essential components should be those that remain invariant across the various demonstrations.

A large body of work explored the use of a symbolic representation to both the learning and the encoding of skills and tasks. The main advantage of a symbolic approach is that high-level skills (consisting of sequences or hierarchies of symbolic cues) can be learned efficiently through an interactive process. However, because of the symbolic nature of their encoding, these methods rely on a large amount of prior knowledge to predefine the important cues and to segment those efficiently. Another body of work focusses on representing the task constraints at a trajectory level to avoid putting too much prior knowledge in the controllers required to reproduce a skill.[1] We follow this approach in our work by using *Gaussian Mixture Model* (GMM) and *Gaussian Mixture Regression* (GMR) to respectively encode a set of trajectories and retrieve a smooth generalized version of these trajectories and associated variabilities. Fig. 1 presents the principles of our approach.

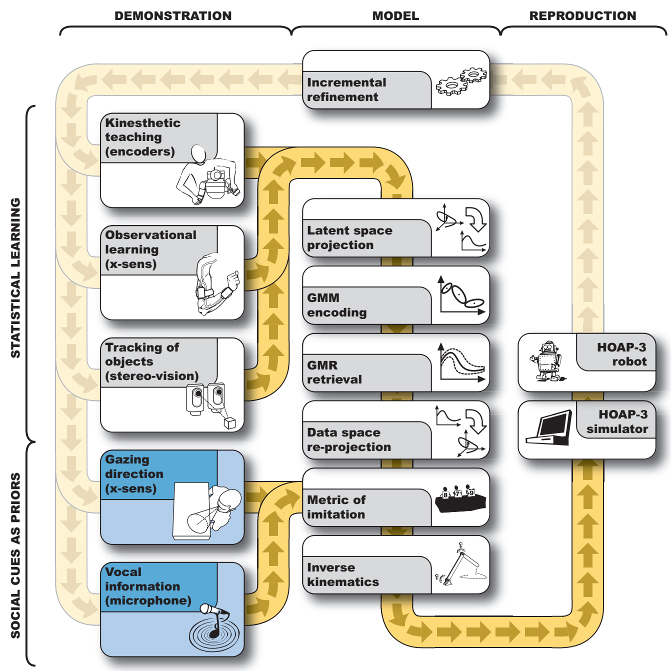


Fig. 1. Information flow across the complete system, where the constraints of a task are extracted statistically through the observation of multiple demonstrations performed in slightly different situations and where various social cues can be used to scaffold the teaching interaction in order to speed up the learning process.

This work was supported by the European Commission as part of the Robot@CWE project (http://www.robot-at-cwe.eu) under contract FP6-2005-IST-5, and as part of the FEELIX GROWING project (http://www.feelix-growing.org) under contract FP6 IST-045169.

S. Calinon and A. Billard are with the Learning Algorithms and Systems Laboratory (LASA), Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland {sylvain.calinon,aude.billard}@epfl.ch

[1] For an exhaustive review and comparisons of the different methods proposed in RPD, the interested reader can refer to [1].

II. Extracting the task constraints through statistical learning

Through the use of GMM, a robot can extract autonomously the essential characteristics of a set of trajectories captured through the demonstrations [3]. GMR can then be used to retrieve a generalized version of the trajectories either in joint space or in task space [2].

Fig. 2 presents the principles of the system. Fig. 3 illustrates the generalization and reproduction methods with an experiment involving manipulation and displacement of objects. In this experiment, the skill is represented as constraints in task space by considering the right hand path relative to two objects tracked by the robot in its environment. The constraints associated with the position of the right hand with respect to an object n are thus represented by the generalized trajectory $\hat{x}^{(n)}$ and associated covariance matrices. We see that by encapsulating the task constraints through GMR, the robot can reproduce the learned skill in new situations (new initial positions of objects).

A. Scaffolding by using different modalities

A trend of research draws the attention on the role of the teacher as being one of the most important key component for an efficient transfer of the skill, where the teaching interaction allows the user to become an active participant in the learning process (i.e. not only a model of expert behaviour). This active teaching process allows the learner to experience and adapt the skill for his/her particular body capacities, as suggested by developmental psychology studies.

In [2], we adopted this strategy and showed that the skill transfer process could benefit from the user's capacity to adapt his/her teaching strategies to the particular context. We presented experiments where a humanoid robot learns new manipulation skills by first observing a human demonstrator (through motion sensors) and then gradually refining its skill through kinesthetic teaching (see Fig. 4). The user thus provides scaffolds to the robot for the reproduction of the skill by moving kinesthetically a subset of the motors. Through the supervision of the user who progressively dismantles the scaffolds after each reproduction attempt, the robot can finally reproduce the skill on its own.

We take the perspective that unlike observational learning, *pedagogy* is required to facilitate the transfer of the skill, which is a special type of communication used to manifest the relevant knowledge of a skill. We thus suggest to use different modalities to produce the demonstrations, similarly to a teaching process where a human teacher would first demonstrate the complete skill to the learner, followed by practice trials performed by the learner under the supervision of the teacher. In our setup, the user can first control simultaneously a large number of degrees of freedom through the motion sensors suit to demonstrate natural gestures. Then, he/she can provide partial demonstrations through kinesthetic teaching (see Fig. 5), i.e. by using the robot's own kinematics in the robot's own environment, which allows him/her to feel the robot's body

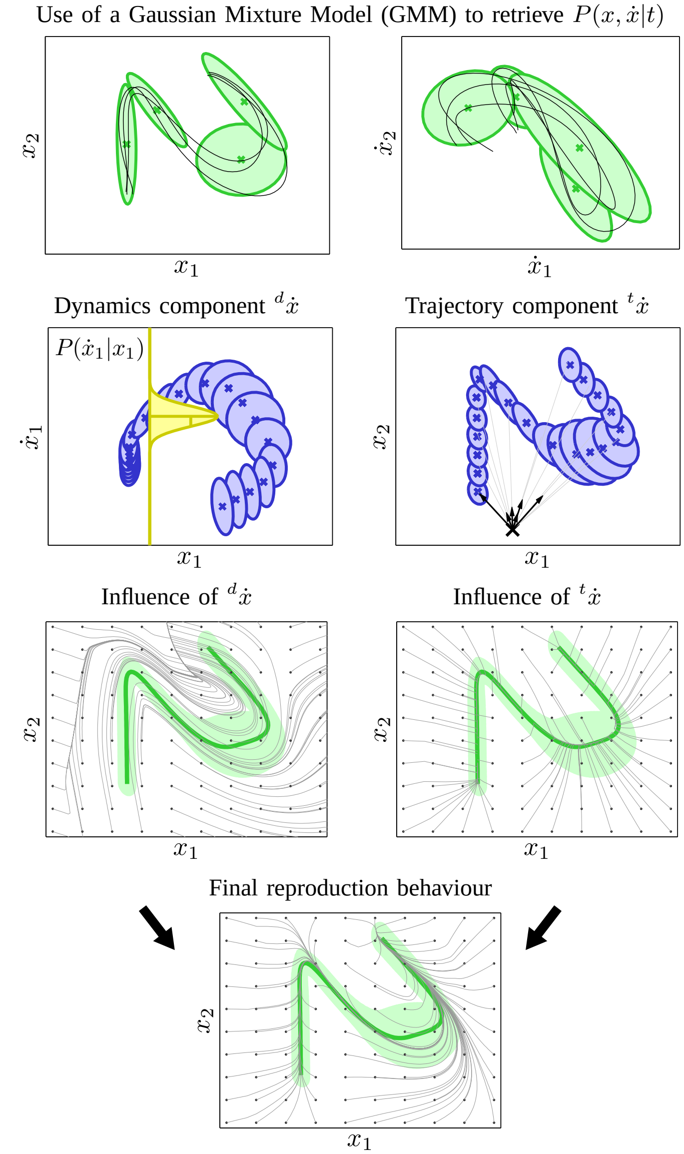


Fig. 2. Illustrative example for the encoding and reproduction of three demonstrations of a motion. *First row:* The set of trajectories $\{t, x, \dot{x}\}$ are first encoded in a Gaussian Mixture Model (GMM), where the components represent respectively temporal, position and velocity values (the motion is represented here only in a 2D plane). *Second row:* Gaussian Mixture Regression (GMR) is then used during the reproduction process to retrieve $P(x, \dot{x}|t)$, which allows to define a dynamics component (*left*) estimating the velocity command required at each iteration to follow the dynamics learned by the system (with respect to the current position), and a trajectory component (*right*) used by the system to come back to a known position in task space (i.e. the learned trajectory is used here as an attractor). *Third row:* Influence of the two velocity commands when used separately and by starting from several initial positions (equally distributed in the workspace). On the one hand, the dynamics component follows the learned motion but tends to become unstable after a few iterations or by starting from an unexplored position. On the other hand, the trajectory component acts as an attractor to the closest point of the generalized trajectory. *Fourth row:* Reproduction behaviour by considering simultaneously at each iteration the influence of the two velocity components. The dynamics component allows to follow the demonstrated dynamics while the trajectory component prevents the robot from moving far away from an unlearned situation and to come back to an already encountered position if a perturbation occurs.

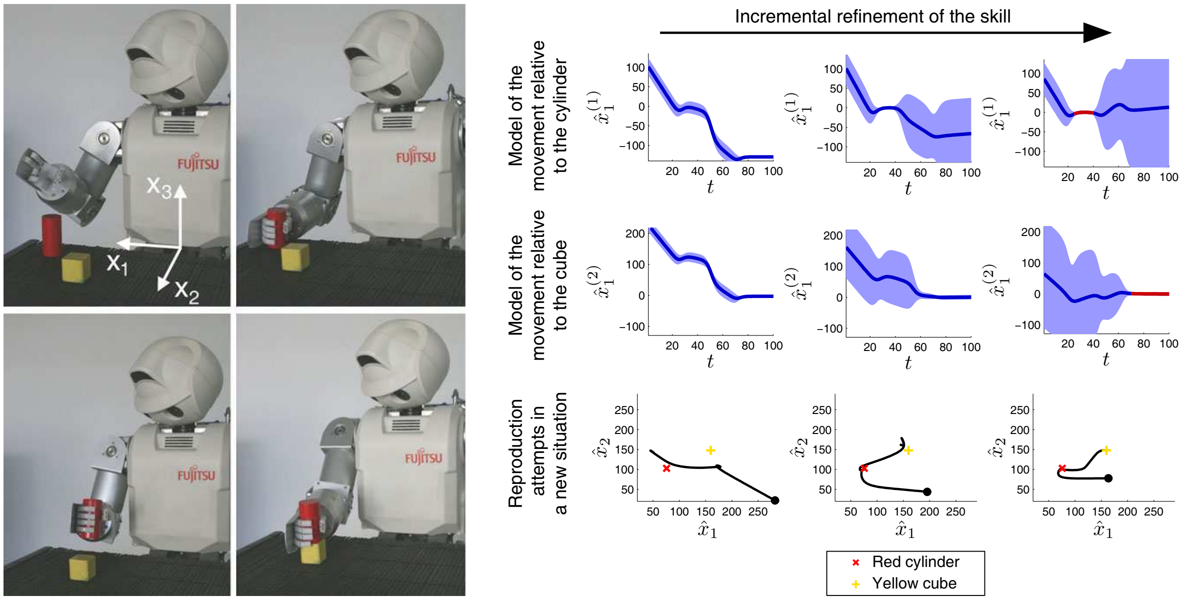


Fig. 3. Incremental refinement of a stacking task that consists of grasping a first object (a cylinder) and putting it on a second object (a cube). The robot learns generalized trajectories coded in frames of reference located on the objects that are manipulated. The three columns of the graph correspond respectively to a representation of the task constraints after 1, 3 and 6 demonstrations. The first two rows show the refinement of the GMR model representing the constraints for the cylinder (*first row*) and for the cube (*second row*) along the movement. After a few demonstrations, the trajectories relative to the two objects are highly constrained for particular subparts of the task, namely when reaching for the cylinder (thin envelope around time step 30) and when placing it on the cube (thin envelope around time step 100). The last row shows the robot's reproduction attempts (after 1, 3 and 6 demonstrations) for a new situation that has not been demonstrated. After 6 demonstrations, the robot correctly reproduces the essential characteristics of the skill, namely reaching for the cylinder and dropping it on the cube (see [2] for a complete description of the results).

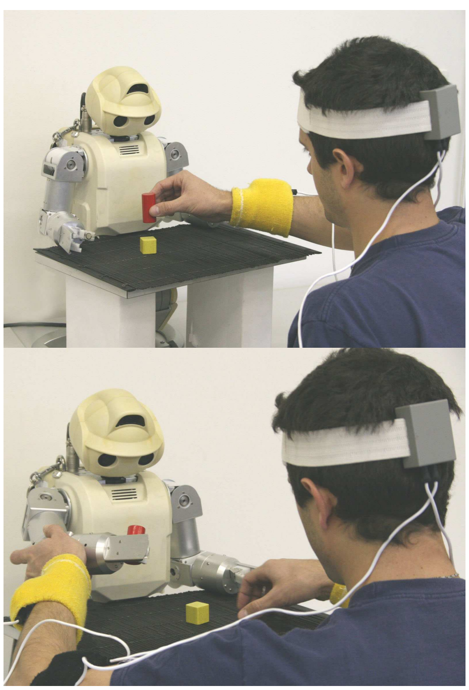


Fig. 4. Different modalities are used to convey the demonstrations and scaffolds required by the robot to learn a skill. The user first demonstrates the whole movement while wearing motion sensors (*top*) and then helps the robot refine its skill through kinesthetic teaching (*bottom*), that is, by grasping the robot's arms and moving them through the motion. The motors are set to passive mode, which allows the user to move freely the corresponding degrees of freedom while the robot executes the task, thus providing partial demonstrations while the robot executes the remaining motion (see [2] for details).

limitations and provide appropriate examples that take these limitations into consideration.

B. Extending the approach to the use of social cues

The system presented above requires to observe the skill in slightly different situations. Even if this variation appears naturally when executing the skill several times, the robot's capacity to generalize over different contexts also depends on the pedagogical quality of the demonstrations provided (e.g. gradual variability of the situations and exaggerations of the key features to reproduce).

This fact shares similarities with the human way of teaching. Indeed, a good teacher also extends the demonstrations progressively so that the learner can more easily infer the connections between the different examples and the range of the possible situations where the skill may apply. In the application presented above, an expert user displaces progressively the objects after each demonstration to provide variability in the exposures of the skill. In such a situation, it is nearly always possible for the robot to extract the task constraints with only a few demonstrations (from four to ten for most of the tasks that we have considered). However, it may happen that untrained users provide a set of demonstrations remaining either too similar or too different from one example to the other.

To weaken the drawback of such situations, we propose to enhance the statistical learning strategy with information coming from various social cues, and show that these cues can be represented statistically as priors in the GMM/GMR framework. We provide two examples with gaze and speech

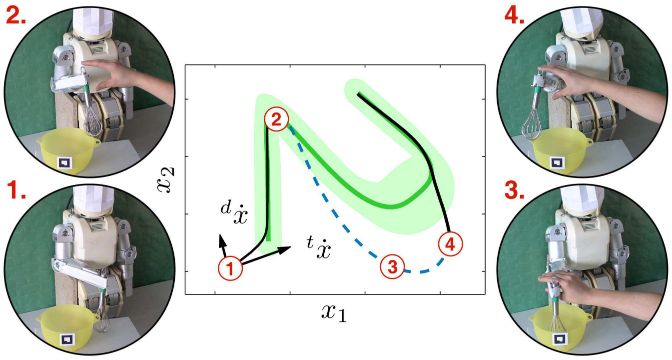


Fig. 5. Illustration of the scaffolding process where the learned motion is represented in thick line with an associated surrounding envelope, resulting from the Gaussian Mixture Regression process described in Fig. 2. *1.* The robot begins to reproduce the learned skill by starting from a new initial position. *2.* At some point during the reproduction, the user holds the robot's arm and provides support for the reproduction of the skill. *3.* The robot lets the user move manually the selected motors (kinesthetic teaching) and records proprioceptive information about its own body motion, while trying to follow the demonstrated motion with the remaining motors that are not controlled by the user. *4.* By releasing the robot's arm, the user then lets the robot pursue the remaining part of the motion on its own. We see here that the robot smoothly comes back to the learned motion. This new demonstration is then used by the robot to refine its model of the skill.

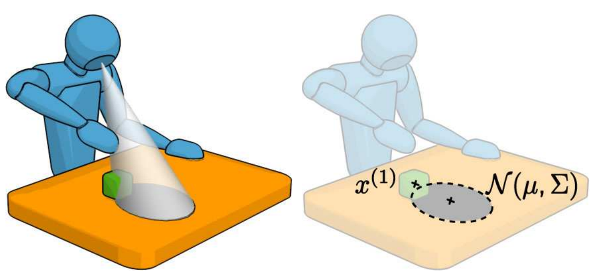


Fig. 6. Estimation of the user's gaze direction during the demonstration of a task as an additional source of information to speed up the learning process. The orientation of the head is recorded through the use of motion sensors (see also Fig. 4). The focus of attention is first estimated by representing the gaze direction as a cone of vision which intersects with the table (forming an ellipse that can also be represented as a covariance matrix). By knowing the position of the objects through the robot's stereoscopic vision system, it is then possible to associate at each time step weighting factors to the different objects observed by the robot in order to highlight the use of these different objects for the particular sub-tasks of the demonstration.

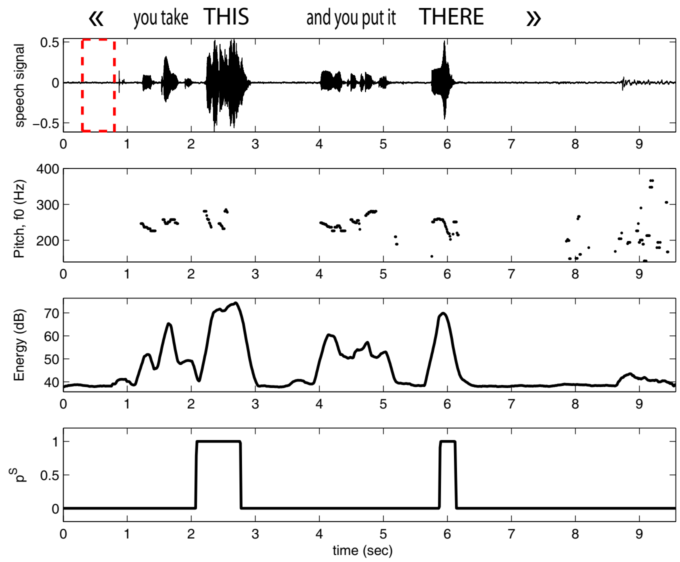


Fig. 7. Extraction of priors from speech for the task depicted in Fig. 3 by extracting attentional events in the vocal trace through pitch and energy information (the temporal window of size W used to detect attentional cues is represented in dashed line). The first row shows the sound signal corresponding to the sentence *"You take THIS and you put it THERE"* told by the user when executing the skill (while observed by the robot, see top snapshot in Fig. 4). We see that the particular events in the demonstration, corresponding respectively to the subparts when the user grasps one object (*"THIS"*) and drops it on the other object (*"THERE"*), are highlighted through the user's voice. These events correspond roughly to local patterns characterized by a higher energy and a larger pitch amplitude with consecutive rising and falling intonation contours, which are typical to prosodic patterns serving as spotlights during the interaction, and which are automatically captured through the Hidden Markov Model encoding. The bottom graph represents the extracted probability p_j^S at time t_j of hearing an attentional utterance.

information. By representing gaze direction as a cone of vision turned towards several objects on a table, the intersection of the cone with the surface can be represented as a Gaussian distribution that can be incorporated easily in the learning framework, see Fig. 6. Similarly, by extracting energy and pitch information from the vocal trace, *Hidden Markov Models* (HMMs) can be used to sort out attentional bids from neutral utterances, see Fig. 7.

These early results show that the integration of social cues within our statistical learning approach is promising. As only a very limited dataset has been used so far, the robustness of the approach still needs to be evaluated with untrained users teaching new skills in real-world experimental setups. One direction of ongoing work is thus to investigate the dependencies and relevance of these different cues in a human-robot teaching interaction context.

Further work will extend the proposed scenarios to more complex interactions where the robot can also refine a learned motion on its own by exploring its environment, and by designing learning scenarios where the teaching phase and reproduction phase are more closely intertwined, allowing richer interactions where the user can provide advices and feedbacks to the robot on its reproduction attempts. Longer-term goals focalize on developing robots that would have the capability to understand and predict the user's intent behind his/her demonstrations, which would for example allow them to learn new skills even from failed attempts.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 1371–1394.

[2] S. Calinon and A. Billard, "What is the teacher's role in robot programming by demonstration? - Toward benchmarks for improved learning," *Interaction Studies*, vol. 8, no. 3, pp. 441–464, 2007.

[3] S. Calinon, F. Guenter, and A. Billard, "On learning, representing and generalizing a task in a humanoid robot," *IEEE Trans. on Systems, Man and Cybernetics, Part B*, vol. 37, no. 2, pp. 286–298, 2007.

Robust Recognition of Complex Gestures for Natural Human-Robot Interaction

Maren Bennewitz*, Tobias Axenbeck*, Sven Behnke†, and Wolfram Burgard*
*Institute for Computer Science, University of Freiburg, D-79110 Freiburg, Germany
†Institute for Computer Science, University of Bonn, D-53117 Bonn, Germany

***Abstract*— Robots coexisting with humans in everyday environments should be able to interact with them in an intuitive way. This requires that the robots are able to recognize typical gestures performed by humans such as pointing gestures, waving, or head shaking/nodding. We present a system that is able to spot and recognize complex, parameterized gestures from data of a monocular camera. To represent people, we locate their faces and hands using trained classifiers and track them over time. We use few, expressive features extracted from this compact representation as input to hidden Markov models (HMMs). First, we segment the gestures into distinct phases and train HMMs for each phase separately. Then, we construct composed HMMs, which consist of the individual phase-HMMs. Once a specific phase is recognized, we estimate the parameter of a gesture such as the target of a pointing gesture. As we demonstrate in the experiments, our system is able to robustly spot and recognize a variety of complex gestures.**

I. Introduction

Robotic assistants designed to communicate with untrained users must be able to interact with them in a natural way. Our humanoid robot (see Fig. 1) is able to generate a variety of natural arm and head gestures that support its speech [1]. When evaluating questionnaires filled out by people who interacted with the robot at former public demonstrations, we discovered that they were confused by the asymmetry between action generation and perception. The robot's visual perception of people was limited to head position and size at that time. To reduce this asymmetry, it is necessary that the robot also recognizes gestures performed by humans. This requires robust and accurate tracking of human body parts as well as the ability to spot and recognize typical gestures in order to infer non-verbal signals of attention and intention.

We present a system that is able to spot and recognize complex gestures from data of a monocular camera. We consider gestures performed with head and arms, such as head shaking/nodding or hand waving as well as parameterized gestures, such as pointing gestures or gestures indicating the size of objects. Figure 2 shows examples of such typical gestures performed by humans during an interaction.

The contribution of our work is a robust and fast gesture recognition method that relies only on data of a monocular camera (no stereo). In contrast to previous approaches relying on monocular image sequences (e.g., [7, 4]), our system works under realistic settings such as varying and difficult lighting conditions, multiple people, and cluttered background. On a notebook computer, we achieve a frame rate of 20 fps and are able to spot gestures as well as to recognize them, i.e., our system distinguishes between previously learned gestures and irrelevant or unconscious movements.

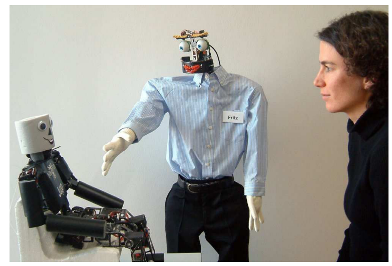
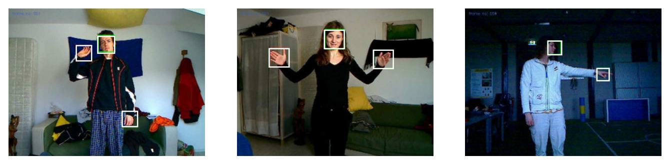


Fig. 1. Our humanoid robot interacts with people using multiple modalities such as speech, facial expressions, eye-gaze, and gestures.



(a) (b) (c)

Fig. 2. Snapshots of typical gestures analyzed in our experiments: (a) waving, (b) indicating the size of an object, and (c) pointing to an object. Our system works robustly even with cluttered background and under different lighting conditions. The bounding boxes highlight detected faces and hands.

Our approach proceeds in three stages. First, we locate faces and hands in the images and update a probabilistic belief which tracks detected faces and hands over time. Second, we extract features from this compact representation of humans. Finally, these features are used as input to Hidden Markov Models (HMMs) which are trained for individual phases of the gestures. Our system recognizes a variety of complex gestures and can estimate their parameters. Existing techniques for parameter estimation of gestures either concentrate on pointing gestures only [3, 5] or rely on the assumption that the whole gesture can be observed [11]. In contrast to that, our approach allows for the estimation of parameters for general gestures once a specific phase is recognized.

II. Representation and Tracking of Humans

For locating faces and hands in the images, we use the object detection framework proposed by Viola and Jones [9] and train reliable and fast classifiers which operate on grey-scale images. To speed-up the search for hands and to increase robustness, we use an adaptive skin color model (which is initially based on the detected face) and constrain the search to skin-colored regions.

We train two kinds of hand classifiers: a *generic* classifier that detects hands and rejects non-hands and a *specific* classifier that is able to discriminate right hands from left ones. Our hand detection system proceeds in two stages. First, the generic hand detector is applied to skin-colored regions. In case it succeeds, the specific hand classifier is applied. In contrast to other approaches [2, 6], our system is able to robustly locate and track hands with a large number of substantially different shapes and to furthermore determine whether a hand is a left or right one.

We maintain a probabilistic belief about the existence of people and the positions of their faces and hands over time. Using this belief, our system improves robustness, can deal with false detections, and is not restricted to a single person.

Additionally, we track the 3D head pose of people. We use an appearance-based approach [8] which locates distinctive facial features. The positions of the features within the face bounding box serve as input to a neural network which computes the three Euler angles of rotation around the neck.

III. Learning and Recognizing Complex Gestures

In our work, we focus on typical gestures performed by humans during an interaction. We currently consider six different types of gestures:

1) *Waving*: One-handed gesture.
2) *Pointing*: Parametric one-handed gesture.
3) *Thisbig*: This parametric two-handed gesture is carried out to indicate the size of an object.
4) *Dunno*: This two-handed gesture is used to express ignorance (informal short for *don't know*).
5) *Head shaking*.
6) *Head nodding*.

A. Gesture Modeling

To model the complex arm gestures *Waving*, *Pointing*, and *Thisbig*, we use three phases: the preparation phase which is an initial movement before the main gesture, the hold phase which characterizes the gesture, and the retraction phase in which the hand moves back to a resting position. Our motivation behind this segmentation is that once the hold phase is recognized, the parameters of *Pointing* and *Thisbig* can be estimated. Furthermore, this segmentation supports the modeling of *Waving* during which similar movements are repeated several times. The less complex gestures *Dunno* and *Head shaking/nodding* are modeled monolithically. We train individual HMMs for each phase of a gesture separately. Accordingly, we train an overall number of 12 HMMs for the gestures/gesture phases.

In our experiments, continuous left-right HMMs with 3-5 (non-skip) states and a mixture of two Gaussians as output distribution performed best to learn the gestures. We use Viterbi training and the Baum-Welch algorithm to estimate for an HMM λ the transition probabilities a^{λ}_{ij} between states i and j and the observation probabilities $b^{\lambda}_{j}(o)$ for a state j given an observation o.

To be able to identify movements not corresponding to any learned gesture, we train an additional model. Here, we follow the approach presented by Yang *et al.* [12] and build a HMM by copying all states from all trained models and arrange them in a fully connected HMM with smoothed output probabilities.

B. Gesture Recognition via Composed HMMs

The gesture phases appear in a specific order which has to be considered during recognition. Fig. 3 illustrates the HMM topology for one- and two-handed gestures as well as for head gestures. As indicated by the arrow, the hold phase can occur several times or last differently long. To identify the most likely gesture given a composed HMM, we apply the Viterbi algorithm [10]. The Viterbi algorithm computes the state sequence with maximum likelihood given an observation sequence $O_{1:T} = o_1, \ldots, o_T$. For the HMM λ, the likelihood of the best state sequence of length t ending in state j is recursively defined as

$$\delta_t(j) \quad = \quad \max_{1 \leq i \leq N^{\lambda}} \delta_{t-1}(i) a^{\lambda}_{ij} b^{\lambda}_{j}(o_t), \delta_1(j) = \pi^{\lambda}_{j} b^{\lambda}_{j}(o_1). \quad (1)$$

Here, a^{λ} and b^{λ} are the parameters of λ, N^{λ} is the number of states, and π^{λ}_{j} specifies the initial state distribution. The algorithm terminates with the computation of the most likely path x^{*}_{T} (which is found via backtracking) and its probability P^{*}

$$P^{*} \quad = \quad \max_{1 \leq i \leq N^{\lambda}} \delta_T(i). \quad (2)$$

In theory, it would be possible to model one- and two-handed gestures in one large HMM. However, to reduce the amount of necessary training data and to improve recognition accuracy, we use separate HMMs for one- and two-handed gestures. Since the HMMs with differently dimensional input features cannot be compared directly, we consider the two-handed HMM if and only if the HMMs for the right and left hand report the same most-likely gesture. This heuristics is applicable since all our two-handed gestures are symmetric.

C. Input Features

As input to the HMMs, we use few, expressive features extracted from the trajectories of head and hands. First, we transform the position of the hands into coordinates relative to the head position and normalize the coordinates with respect to the size of the face bounding box. For one-handed gestures, we use polar coordinates in the image with the head as origin and the velocity. Accordingly, the feature vector $\mathbf{f}_{one}$ is defined as

$$\mathbf{f}_{one} = (r, \phi, v). \quad (3)$$

Here, r is the distance of the hand to the head, ϕ is the angle, and v is the velocity.

Since the two-handed gestures we consider are symmetric, we measure the difference in x/y-direction of their left and right hand coordinates $(x^{l/r}_t, y^{l/r}_t)$ at time t in the features $d_x = |x^l_t| - |x^r_t|$ and $d_y = y^l_t - y^r_t$. Furthermore, we record the sum of the y-coordinates of the hands in the

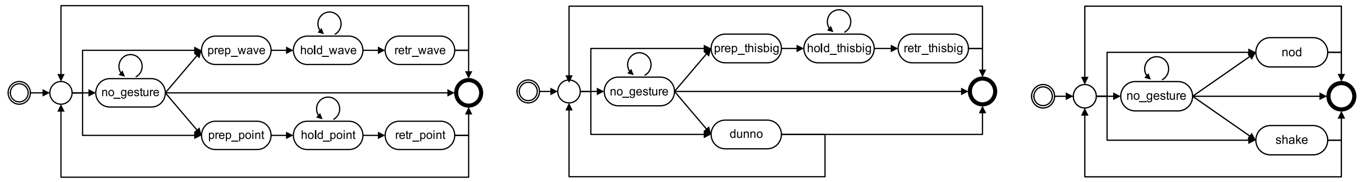


Fig. 3. Composed HMM consisting of the individual phase-HMMs. The first two for one- and two-handed gestures, and the right one for head gestures.

feature $y^{lr} = y^l_t + y^r_t$ and consider the change of the hand coordinates in x-direction

$$\Delta x^l x^r = |x^l_t| - |x^l_{t-1}| + |x^r_t| - |x^r_{t-1}|. \tag{4}$$

As a final feature, we consider the velocities of the hands $v^{lr} = v^l_t + v^r_t$. Thus, the feature vector $\mathbf{f}_{two}$ is defined as

$$\mathbf{f}_{two} = (d_x, d_y, y^{lr}, \Delta x^l x^r, v^{lr}). \tag{5}$$

The head gestures nodding and shaking are described by a feature vector $\mathbf{f}_{head}$ which consists of the three Euler angles of rotation roll, pitch, and yaw as well as their velocities

$$\mathbf{f}_{head} = (\theta^r, \theta^p, \theta^y, v^{\theta_r}, v^{\theta_p}, v^{\theta_y}). \tag{6}$$

D. Estimating Parameters of Gestures

Currently, we consider two parameterized gestures: *Thisbig* and *Pointing*. The corresponding parameters are estimated during the hold phase of the respective gesture. For *Thisbig*, the estimation is done straightforwardly using a learned mapping to estimate the distance of the person to the camera given the bounding box size of the face.

For the estimation of pointing targets, we use of the three rotation angles of the head pose. We assume that people are looking to the object of interest they want to draw the attention to and that the head pose coincides with the gaze direction. Furthermore, we assume the 3D positions of potential pointing targets to be known. First, we estimate the 3D position of the head using the above mentioned mapping from bounding box size to distance. Starting from that position, we construct a straight line in direction of the roll, pitch, and yaw angle of the head pose. Finally, we determine the object which has the closest distance to that line.

IV. Experiments

We performed a series of experiments in order to evaluate our approach. To collect training data, we asked five different people to perform gestures in a distance of 1.5-2.5m to the camera. We chose two different locations, different lighting conditions, and different backgrounds (see Fig. 2). We recorded and processed the videos with a rate of 20fps and used a resolution of 640×480 pixel. We had a database consisting of 75 samples per gesture which we manually labeled, i.e., we marked the start and the end of each gesture as well as the beginning and end of the hold phase.

A. Gesture Recognition

After training the phase-HMMs for the hand gestures, we tested their ability in distinguishing the individual gesture phases (preparation (p), hold (h), and retraction (r) phase). We used the Viterbi path and counted the number of correctly recognized gesture phases from the number of all test sequences. Tab. I shows the percentage of correctly recognized segments for one-handed gestures. As can be seen, using the extracted features, the individual phases of one-handed gestures can correctly be recognized. Only one error occurs for a segment containing a *retr_point* phase which is classified as *retr_wave*. This can be explained by the fact that both retraction phases contain similar movements in the end. When considering a whole observation sequence consisting of all three phases, this error does not occur since the preparation and hold phase are correctly recognized. For the recognition of two-handed gestures shown in Tab. II, it can be seen that in a single test sequence, the phases of *Thisbig* are classified as *Dunno*. When sequences in which persons are not performing any gesture are included into the test set, we achieve an overall recognition rate of 90% for one- as well as for two-handed gestures. The largest part of this error results from the fact that it sometimes happens that *no_gesture* phases are classified as the preparation phase of a gesture.

The following experiment is designed to evaluate the performance of our system on sequences containing whole gestures. We computed the Viterbi path in the composed HMMs at each time step and counted how often the most likely hypothesis corresponds to the true gesture. Fig. 4 shows the results for all six gestures. As can be seen, the gestures can be reliably recognized after processing only few frames. Nodding seems to be most difficult to recognize because sometimes people barely move their head. And, again, we made the observation that *Thisbig* sometimes tends to be classified as *Dunno*.

To better evaluate the ability of our HMMs to distinguish arm gestures, we performed experiments in which we computed for a given observation sequence the Viterbi path and its likelihood for all individual gesture HMMs consisting of the corresponding phase-HMMs (i.e., we did not use the composed HMMs here). We then computed the joint probability $P(g^l, g^r)$ of the gesture g^l of the left and the gesture g^r of the right hand. Fig. 5 plots the evolution of the probabilities of the gestures over time for a sequence in which a person waved with the left hand. In the beginning, the person was not performing any meaningful gesture and, thus, the *no_gesture* model had the highest probability. Afterwards, the probability of the correct gesture increased.

TABLE I

RECOGNITION OF ONE-HANDED GESTURE PHASES.

	p_wave	*h_wave*	*r_wave*	*p_point*	*h_point*	*r_point*	*rec. rate*
p_wave	25	0	0	0	0	0	100%
h_wave	0	25	0	0	0	0	100%
r_wave	0	0	25	0	0	0	100%
p_point	0	0	0	25	0	0	100%
h_point	0	0	0	0	25	0	100%
r_point	0	0	1	0	0	24	96%

TABLE II

RECOGNITION OF TWO-HANDED GESTURE PHASES.

	dunno	*p_thisbig*	*h_thisbig*	*r_thisbig*	*rec. rate*
dunno	25	0	0	0	100%
p_thisbig	1	24	0	0	96%
h_thisbig	1	0	24	0	96%
r_thisbig	1	0	0	24	96%

B. Parameter Estimation

Finally, we asked people to point to predefined targets. We positioned eight different targets within a range of 1.5m to the camera and at different heights. The hold phase of all 66 pointing gestures was identified and the correct target was estimated in 80% of all cases.

Second, we asked people to indicate the size of objects. We told them to indicate the sizes 25cm, 50cm, 100cm, and 150cm and estimated the parameter in the hold phase. We performed 32 experiments and counted the nearest neighbor class of each estimate. Our system was able to determine the correct class in 94% of all cases.

C. Videos

Illustrating videos can be found at our web page[1]. The videos show the robustness of our approach to recognize complex gestures performed by different people. As the experiments demonstrate, gestures can reliably be recognized even under varying lighting conditions and with cluttered background.

V. CONCLUSIONS

We presented an approach to robustly recognize gestures from data of a monocular camera. We consider typical gestures performed by humans during an interaction such as nodding or pointing. To represent people, we locate and track their heads and hands. We use few, expressive features extracted from this compact representation as input to HMMs. We segment complex gestures into three phases and train HMMs for each phase separately. We then construct HMMs composed of the individual phase-HMMs. Using the distinction between different phases, we are able to estimate parameters of gestures as soon as a certain phase is recognized.

Our approach has been implemented and evaluated on a humanoid robot. As the experimental results show, our system is able to reliably spot and recognize gestures, i.e., it distinguishes between previously learned gestures and irrelevant or unconscious movements.

[1]http://www.informatik.uni-freiburg.de/˜maren/animations-gestures.html

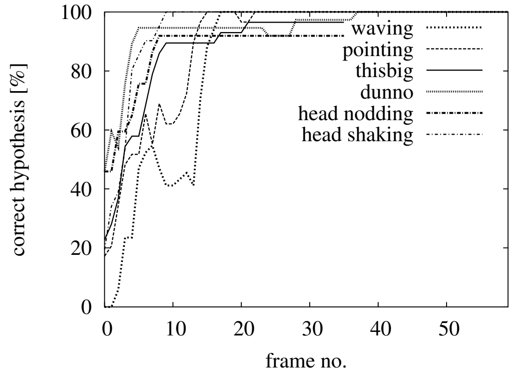


Fig. 4. Number of frames after which the most likely hypothesis is the correct gesture.

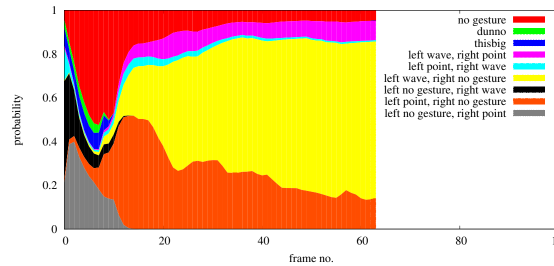


Fig. 5. Evolution of the probabilities of the gestures over time for an experiment in which a person waved with the left hand.

ACKNOWLEDGMENT

This project is supported by the DFG, grant BE 2556/2-2 and by the BMBF project DESIRE.

REFERENCES

[1] M. Bennewitz, F. Faber, D. Joho, and S. Behnke. Fritz – A humanoid communication robot. In *Proc. of the IEEE Int. Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2007.

[2] M. Kolsch and M. Turk. Robust hand detection. In *Proc. of the Sixth IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG)*, 2004.

[3] C. Martin, F.-F. Steege, and H.-M. Gross. Estimation of pointing poses for visual instructing mobile robots under real-world conditions. In *Proc. of the 3rd European Conference on Mobile Robots (ECMR)*, 2007.

[4] J. A. Montero and L. E. Sucar. Feature selection for visual gesture recognition using hidden Markov models. In *Proc. of the Fifth Mexican Int. Conf. in Computer Science*, 2004.

[5] K. Nickel and R. Stiefelhagen. Detection and tracking of 3D-pointing gestures for human-robot-interaction. In *Proc. of the IEEE/RSJ Int. Conf. on Humanoid Robots (Humanoids)*, 2004.

[6] E.-J. Ong and R. Bowden. A boosted classifier tree for hand shape detection. In *Proc. of the Sixth IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG)*, 2004.

[7] G. Rigoll, A. Kosmala, and S. Eickeler. High performance real-time gesture recognition using hidden Markov models. In *Proc. of the Int. Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*, 1998.

[8] T. Vatahska, M. Bennewitz, and S. Behnke. Feature-based head pose estimation from images. In *Proc. of the IEEE/RSJ Int. Conf. on Humanoid Robots (Humanoids)*, 2007.

[9] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[10] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. on Information Theory*, 13(2):260–269, 1967.

[11] A.D. Wilson and A.F. Bobick. Parametric hidden Markov models for gesture recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(9):884–900, 1999.

[12] H.-D. Yang, A.-Y. Park, and S.-W. Lee. Robust spotting of key gestures from whole body motion sequence. In *Proc. of the 7th IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG)*, 2006.

Learning new behaviors : Toward a Control Architecture merging Spatial and Temporal modalities

Matthieu Lagarde*, Pierre Andry*, Giovannangeli Christophe* and Philippe Gaussier*†
Email: {lagarde,andry,giovannangeli,gaussier}@ensea.fr
*Neurocybernetic team, ETIS Laboratory, CNRS UMR 8051, Univ of Cergy-Pontoise, ENSEA, Pontoise 95000, France
†Institut Universitaire de France

***Abstract*— This paper discusses the role of two antagonist neural networks for the learning and control of complex behaviors composed as a sequence of elementary states. Learning a pathway with a mobile robot or a sequence of actions with a robot arm can be seen either as the result of the learning of a temporal sequence or as the result of the natural dynamics of a sensory-motor system using appearance based approaches for instance. As a result, we will discuss the performances and the complementary features of each system, and propose a unique control architecture embedding both systems for long life learning.**

I. Introduction

Our long term goal is to design a control architecture allowing a robot to learn, as autonomously as possible, sequences of actions related either to spatial or temporal constraints (displacements between places or gestures for instances). Learning a behavior is often related to learning by reinforcement, by demonstration or learning by imitation. Learn by imitation has often been considered as a complex behavior, but in previous work we have showed that the imitation can emerge from elementary mechanisms. For example: a robot that learns a "behavior" consisting in moving at different places and performing some very simple but different manipulations of different objects at each places as shown in figure 1. This

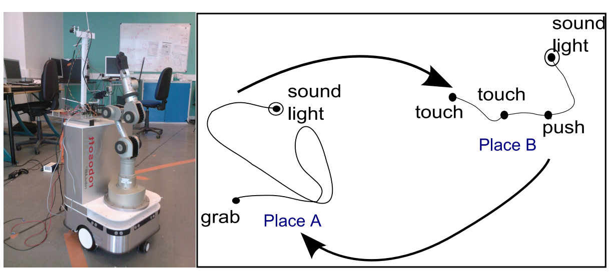


Fig. 1. Scenario illustrating our long term goal.

objective raises the issue of learning a behavior composed of actions : the nature of the relevant information is different between "moving from a place to another", and "using an arm to push an object". Indeed, the working spaces, the type of inputs, the motor commands, are different.

In the field of navigation systems, learning a sequence of displacements between places (also known as navigation and planning) is strongly related to the localization and mapping (see for example the SLAM literature). The sequence is generally the result of a plan composed of motor actions or the result of an imitation (wheel orientation and speed) associated to the recognition of places (localization) anchoring the behavior in the robot's cognitive map.

In the field of "manipulating systems", i.e. non-mobile systems performing gestures and/or object manipulation. Different models propose to learn and adapt motor trajectories of the mechanical system in order to fit with the desired one of the model. The sequence is strongly related to the dynamical parameters allowing shaping the trajectory of the arm's joints in order to obtain the right reproduction of the behavior.

This very short presentation of two important fields of autonomous robotics illustrates how complex the issue of building a global system that deals with navigation and arm movements as a single problem is. Our approach implicitly raises two crucial questions : how to build control architecture for articulated and mobile robots (to consider manipulation and navigation as a single problem)? How to build a neural architecture for spatial and temporal sensory-motor learning in which each modality could complement, confirm, infirm and/or enrich the other? What are the minimal requirements for such a merging? Which level for fusion making? Which coding to employ? In order to start to answer to these issues, we compare two models in the purpose of a unified model. Both solutions are based on artificial Neural Networks (NN) inspired from different properties of the cerebellum and the hippocampus loop.

II. Models

Complex Temporal Sequences. The model allows a robot to learn a sequence as a succession of transitions between the different sensory-motor situations. An associative learning rule allows learning and predicting the timing of the transitions. Moreover, neural oscillators composed of coupled CTRNN [Beer, 1994], play the role of an internal context and provide additional information in order to remove ambiguities in complex sequences [Lagarde et al., 2007]. Applied to the navigation, the sequence is based on the succession of

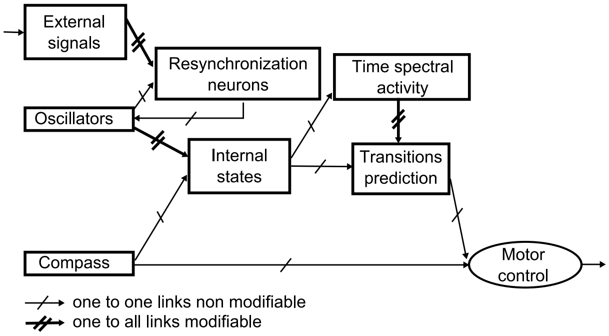


Fig. 2. Model of complex temporal sequences or orientations.

orientations (orientation is obtained from the compass). One of the main problems was the time spent by the robot to turn delaying the perception of the orientation during the reproduction. During this time lag, the internal context (i.e. the activities of the oscillators) changes. Consequently, the system loses the internal state and fails to reproduce the sequence. In order to avoid this problem, we propose to resynchronize the oscillators, according to external signals, when a new internal state is learnt. The context can be associated according to a Least Mean Square (LMS) learning rule. The property of resynchronization is crucial so that the system is able to correctly reproduce a sequence. This property is close to the one used in others models like the Echo States Network (ESN) [Jaeger, 2001].

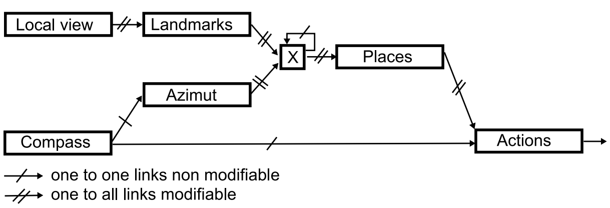


Fig. 3. Model of associations between places and actions.

Association between places and actions. This model [Giovannangeli and Gaussier, 2007] associates places with actions (figure 3). A place is a constellation of visual features (landmark, azimuth). The constellation results from the merging of "what" information provided by the visual system that extracts local-view centred on points of interest. The "where" information provided by the compass. A simple associative learning between places and actions enables to generate a sensory-motor attraction basin for homing or path following behaviors.

III. Robotic Application

The robot used in our experiments is a Robulab10 (Robosoft) with a pan-tilt video camera and a compass.

Association between places and actions. During the learning, the robot moves in the environment (figure 4). When the robot escapes too far from the desired trajectory, we correct it with a joystick as a dog guided with a leash. At this moment, the NN learns online a new association between the correct of motor command and the current place. When the robot recognizes a place, it triggers the associated action. After 3 or 4 iterations of learning, the robot navigates autonomously without correction from the teacher. The system does not try to recognize a place, but use a competitive mechanism between the learnt associations to build an attraction basin. The system adapts to the dynamic of the environment (obstacles, others agents).

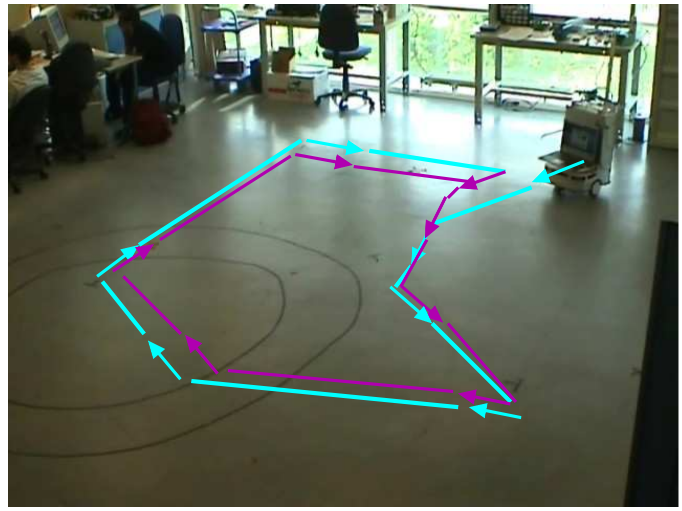


Fig. 4. Spatial navigation: picture of learnt (light arrows) and reproduced (dark arrows) trajectories by the robot with places-actions associations.

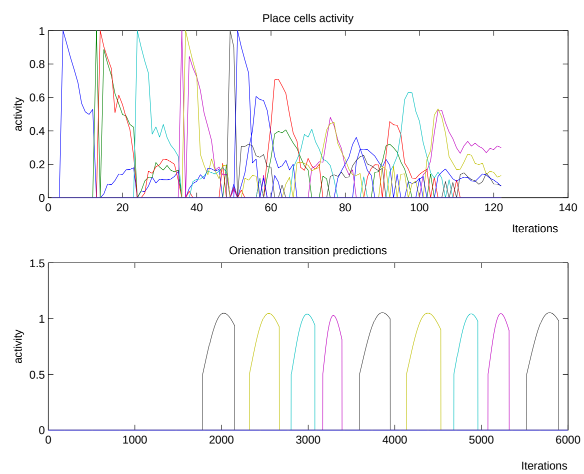


Fig. 5. Each curves represent the activity of each neuron coding place (Up) or transition prediction (Down). Up : responses of the place cells during learning (from iteration 0 to 55) and reproduction (from iteration 55) allowing the triggering of each action of the trajectory. Bottom: Activity of the transition prediction group, allowing the reproduction of the sequence of orientations with the correct timing

Complex Temporal Sequences. During the learning, the robot moves in the environment as shown in figure 6. When the robot makes mistakes, we use a joystick to correct its trajectory. At this moment, the NN learns online (one shot learning) a new transition between the previous and the new orientation. To initiate the sequence of displacement, we set the robot heading to the first learnt orientation (the robot moves at a constant speed). The orientation information triggers on time the prediction of the next orientation that will drive the robot's new rotation, and begin the step by step reproduction of the sequence : each new orientation is recognized and resynchronizes the oscillators, inducing the next prediction and the realization of the associated action.

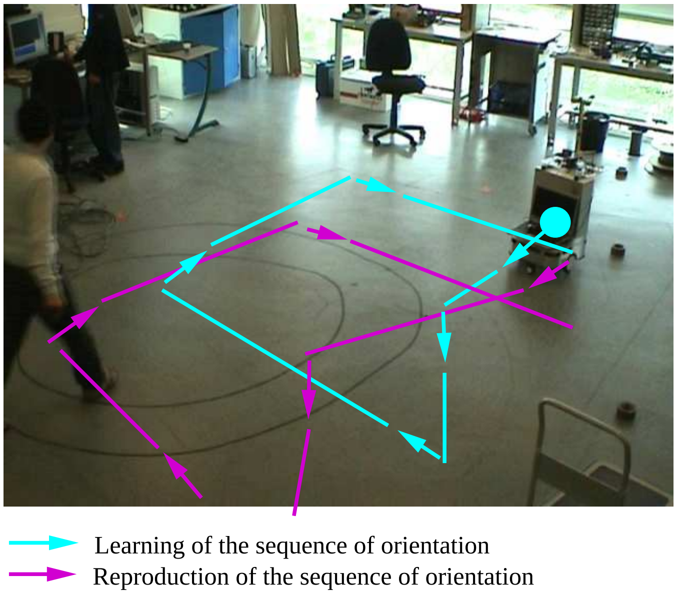


Fig. 6. Learning and reproduction of temporal sequences: picture of another learnt trajectory (light arrows) and reproduced trajectory (dark arrows) by the robot.

IV. Discussion

The architecture that learns places-actions associations has shown to be robust and reliable. It allows the robot to successfully navigate indoor as well as outdoor. In parallel, learning sequences of orientations has been successfully used in previous works in the frame of imitation with mobiles or articulated robots. Of course, the robustness of the navigation is strongly dependent on the quality of the visual environment. If the visual mechanism has shown to be robust to partial changes of the environment, a failure of the camera or very bad lighting conditions will prevent the system from working. Considering this, learning the sequence of orientations for a simple navigation task becomes interesting. Indeed, the robot uses little information from the environment : only a detection of the orientation variations. During the reproduction of the sequence, the robot acts as a "blind" automata. It can work correctly during little iteration without visual information. Resynchronization of internal dynamics with the current state is necessary after a while. It can not adapt to sudden changes of the environment (e.g. a new obstacle). Nevertheless, we think that the models of places-actions associations and sequence learning should work in parallel. Each architecture seems to complete the other one in order to learn the spatial and temporal properties of complex behaviors. Moreover, the learning of the timing of the orientation changes should (1) contribute to confirm or infirm the visual place recognition (being the right orientation at the right time), (2) punctually replace the place cells if their activity is not strong enough (bad visual condition, conflict between different places) and (3) contributes to build long sequences, allowing to concatenate behaviors composed of displacements with those made of sequences of manipulations. Neurobiological and psychological studies suggest that both types of learning cohabit in the brain of mammals. For example, The results in [Packard and McGaugh, 1996] show the different roles of the hippocampus and basal ganglia in the task learning with different learning scales and learning rates, and the implication of different modalities (visual vs. proprioceptives). Hence, we propose a new architecture in Figure 7. This new architecture shows the model connecting

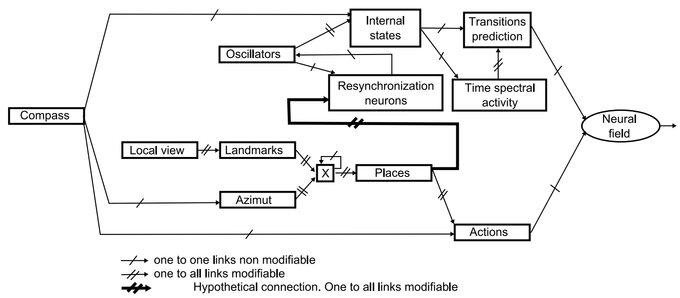


Fig. 7. Proposition of unified model with a hypothetical connection (bold arrow) could allow re-synchronizing the internal context (i.e. the oscillators) on place recognition signals.

both predictors. In order to merge the predictions, both systems will have their outputs merged in one neural field [Amari, 1977], [Schöner et al., 1995] allowing the cooperation of the predictions in case of similar responses, but also their competition in the case of too different responses (capacity of bifurcation of the neural field). Moreover, the neural field will allow coping easily with two systems working at different time scales. The emerging behavior will be the result of two subsystems having different dynamics and categorizing, predicting complementary modalities.

In a future robotic experiment, this new model will also help to enhance human/robot interaction allowing a mobile robot to learn the navigation path directly from following a naive user. During the displacement, the robot will focus on the demonstrator and will learn online the temporal succession of its orientations (short time learning). To anchor the displacement in the environment (which is not possible when focusing on the naive user), the robot will reproduce by oneself the displacement (i.e. the sequence of successive orientations) and will learn during this reproduction the associations between places and actions (long time learning). This experiment would help to study how experiences are stored in the brain. Moreover, it would help to study how and why the brain needs to use different kinds of memories according to learn and store behaviors between the episodic memory (hippocampus) and the long term memory of the know-how (basal ganglia and/or cortical structures).

In the purpose of using an arm mounted on the mobile robot, it is interesting to anticipate that a similar visual mechanism as the “places cells” could guide the arm (for example, the location of interesting visual objects). This mechanism could allow anchoring in the visual working space of the arm temporal sequences of gestures, as well as the navigation model anchors actions in the wide visual environment. Previous works on robot arms have show the importance of the visuo-motor learning for gesture imitation. This solution consists in learning on a multi-modal map the associations between the motor and the visual information of the end-effector.

Acknowledgment

This work is supported by the French Region Ile de France, the FEELIX GROWING European project (FP6 IST-045169), the French Direction Générale des Armées (DGA) and CNRS Neuroinformatique project.

References

[Amari, 1977] Amari, S. (1977). Dynamic of pattern formation in lateral-inhibition type by neural fields. *Biological Cybernetics*, 27:77–87.

[Beer, 1994] Beer, R. (1994). On the dynamics of small continuous–time recurrent networks. Technical Report CES–94–18, Cleveland, OH.

[Giovannangeli and Gaussier, 2007] Giovannangeli, C. and Gaussier, P. (2007). Human-robot interactions as a cognitive catalyst for the learning of behavioral attractors. In *16th IEEE on Robot and Human Interactive Communication 2007*, pages 1028–1033.

[Jaeger, 2001] Jaeger, H. (2001). The ”echo state” approach to analysing and training recurrent neural networks. GMD Report 148, GMD - German National Research Institute for Computer Science.

[Lagarde et al., 2007] Lagarde, M., Andry, P., and Gaussier, P. (2007). The role of internal oscillators for the one-shot learning of complex temporal sequences. In *ICANN 2007*, volume 4668 of *LNCS*, pages 934–943.

[Packard and McGaugh, 1996] Packard, M. and McGaugh, J. (January 1996). Inactivation of hippocampus or caudate nucleus with lidocaine differentially affects expression of place and response learning. *Neurobiology of Learning and Memory*, 65:65–72(8).

[Schöner et al., 1995] Schöner, G., Dose, M., and Engels, C. (1995). Dynamics of behavior: Theory and applications for autonomous robot architectures. *Robotics and Autonomous Systems*, 16(4):213–245.

Moving from augmented to interactive mapping

O. Booij, B. Kröse
Intelligent Systems Lab Amsterdam, University of Amsterdam
1098 SM Amsterdam, The Netherlands
Email: obooij@science.uva.nl

J. Peltason, T. Spexard, M. Hanheide
Applied Computer Science, Bielefeld University
33594 Bielefeld, Germany
Email: jpeltaso@techfak.uni-bielefeld.de

I. Introduction and problem statement

Recently[1] there has been a growing interest in human augmented mapping[1, 2]. That is: a mobile robot builds a low level spatial representation of the environment based on its sensor readings while a human provides labels for human concepts, such as rooms, which are then augmented or anchored to this representation or map [3]. Given such an augmented map the robot has the ability to communicate with the human about spatial concepts using the labels that the human understand. For instance, the robot could report it is in the "kitchen", instead of a set Cartesian coordinates which are probably meaningless to the human.

Even if the underlying mapping method is perfect, two main problems occur in the approach of augmented mapping. When guiding a robot through a number of rooms, humans tend to not provide labels for every visited room [4]. The result is that the robot has difficulty to model where one room ends and the other room starts. This problem could be solved by detecting room transitions through the sensor data. Although good attempts using such an approach have been made in office environments [5, 6], applying these to other environments such as real homes is nontrivial. Another problem is that the generalization of the labeled map to newly acquired sensor data can be much different from the humans ideas. That is: there is a mismatch between the human representation and the robots representation. In our case the robots generalizes labels using visual similarities, while humans could use the function of the room. Even among humans there are differences between spatial representations. Think of a living room with an open kitchen. Where does the living room end and the kitchen begin?

Our solution to both of these problems is to use pro-active human robot interaction. We briefly describe how the robot learns a map of the environment using a vision sensor and active dialog with a human guide. The method is implemented on Biron (the Bielefeld Robot Companion) which supports an integrated human robot interaction system based on XCF (XML Communication Framework) complete with person attention, spoken dialog, person following, gesture recognition and localization components [7].

[1]The work described in this paper was conducted within the EU FP6-002020 COGNIRON ("The Cognitive Companion") project.

Fig. 2. Biron and human guide in a home environment.

II. Augmented mapping

A. Appearance based topological mapping

To map the environment we use images taken by an omni-directional vision system. From each image SIFT features are extracted which are used to find image point correspondences between pairs of images by matching their SIFT descriptors. False point correspondences are then removed by imposing the epipolar constraint. By dividing the minimal number of SIFT features of two images i and j by the number of correspondences, one finds a measure for the distance of the two images in appearance space:

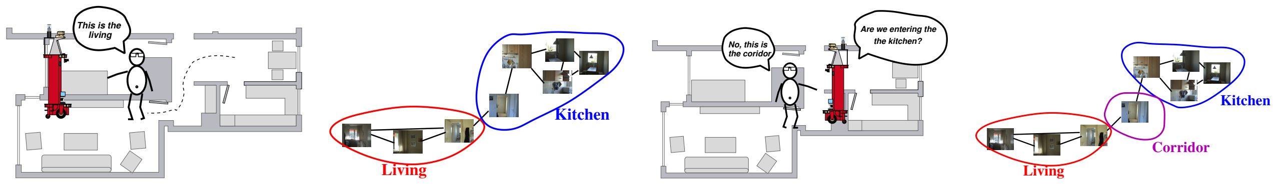
$$d_{ij} = \frac{\min(\#\text{SIFTS}_i, \#\text{SIFTS}_j)}{\#\text{correspondences}_{ij}}$$

These computed distances are put in a graph representation in which the nodes denote the images and distances are put on the links, effectively creating a topological map of the environment. If the distance is above a certain threshold, which was set to 10 in our experiments then no link was created.

The complete map building system is running in real time on one of the robot-laptops, processing around one image per second. To keep the number of comparisons limited we used the Connected Dominating Set method to pick key images from the previous image set. For an in depth treatment of this map building scheme see [8].

B. Human augmentation of room labels

While the robot is driving through the environment following the human guide and building a topological map, room-



(a) (b) (c) (d)

Fig. 1. A sketch of the proposed method. (a) The human guide provides a label. (b) After a second label is provided the map consists of two subgraphs. (c) The robot reports a room transition on which the human provides feedback. (d) The feedback is used to update the map.

labels can be provided to the robot, see Figure I for an example. This is performed by commanding the robot to stop and telling the robot the name of the room it is in, e.g. "This is the kitchen" (see Figure 1(a)). To handle miscommunication, a powerful grounding-based dialog system is used that can handle complex conversational repair behavior and facilitate a smooth conversation (see [2] for more information). The given label is then added to the next node (image) that is added to the map.

Using the given labels and the structure of the graph the robot can partition the map in different subgraphs. Every node is assigned to that label corresponding to the closest labeled node computed with Dijkstra's shorter path algorithm[9] (see Figure 1(b)). Effectively we are exploiting here the fact that images taken in a convex space, which usually correspond to the notion of rooms, are visually much more similar than images taken while the robot moved through a narrow passage, a door.

III. Interactive Mapping

As can be seen in Figure 1(b) the transition from the "living room" to the "dining room" is probably not learned in the way the human had in mind when giving the labels. The human would probably not notice this until it would send the robot to the "Living room" after which the robot would move to the hallway. This can easily be solved by making the robot pro-actively interact with the human.

Every time the robot adds a new image to the map it computes its corresponding label. If this label is different than the label of the previously added node, the robot reports this to the human in the form of a question. In the case of Figure 1(c) the robot asked "We just entered the living room, right?". The human now has the opportunity to provide feedback, possibly reducing the mismatch with its own representation, see Figure 1(d). If later the robot would really enter the "living room" it will again report this to the human confirming that it has correctly learned the transition.

A technical detail is that the robot does not stop driving while reporting room change to the human, so to not interrupt the tour. Thus new nodes are added to the graph while it awaits an answer. The possibly corrected label is put on the node which triggered the robot. This could lead to race conditions if there are a lot of transitions close to each other, e.g. if different locations in the room are also labeled. In the conducted experiments, however, we did not experience such problems.

IV. Results

The new interactive mapping approach was recently implemented on the Biron robot. First test trials were performed in a rented apartment at Bielefeld which was furnished to look like a real home environment. See `http://www.science.uva.nl/˜obooij/research/mappingHRI/index.html` which features a video shot during one of the trials illustrating the capabilities of the complete interactive mapping system.

The robot captured panoramic images once every 2 seconds and the tour took around 5 minutes resulting in a total set of 158 images. The complete mapping system, including the image processing, is performed during the tour in real-time on one of the laptops attached to the robot.

In Figures 3(a)-(e) the spatial representation is plotted using hand-corrected odometry data. Note, however, that this odometry data was not used by the mapping algorithm.

In Figure 3(a) the robot drove from the living room at the bottom right of the figure through the hallway to the kitchen on the upper left. By then the only label that was given was in the living room, so it groups every new node with that label. In Figure 3(b) it is provided a new label "Dining room" and as can be seen the graph is split into two groups according to their distance over the graph. The cut between these two groups is located somewhere inside the small hallway. This became apparent to the guide in Figure 3(c) where the robot was guided back to the hallway and asked if it reentered the kitchen. After interacting with the guide the label "Hallway" was added to the map, splitting the graph in three parts, see Figure 3(d). After reentering the living room the robot again asked if this was the "Living room" which was confirmed by the guide resulting in another node being labeled. In Figure 3(e) the final spatial representation is shown as build by the robot.

V. Conclusion

We have shown that using relatively simple human robot interaction techniques we can solve two problems apparent

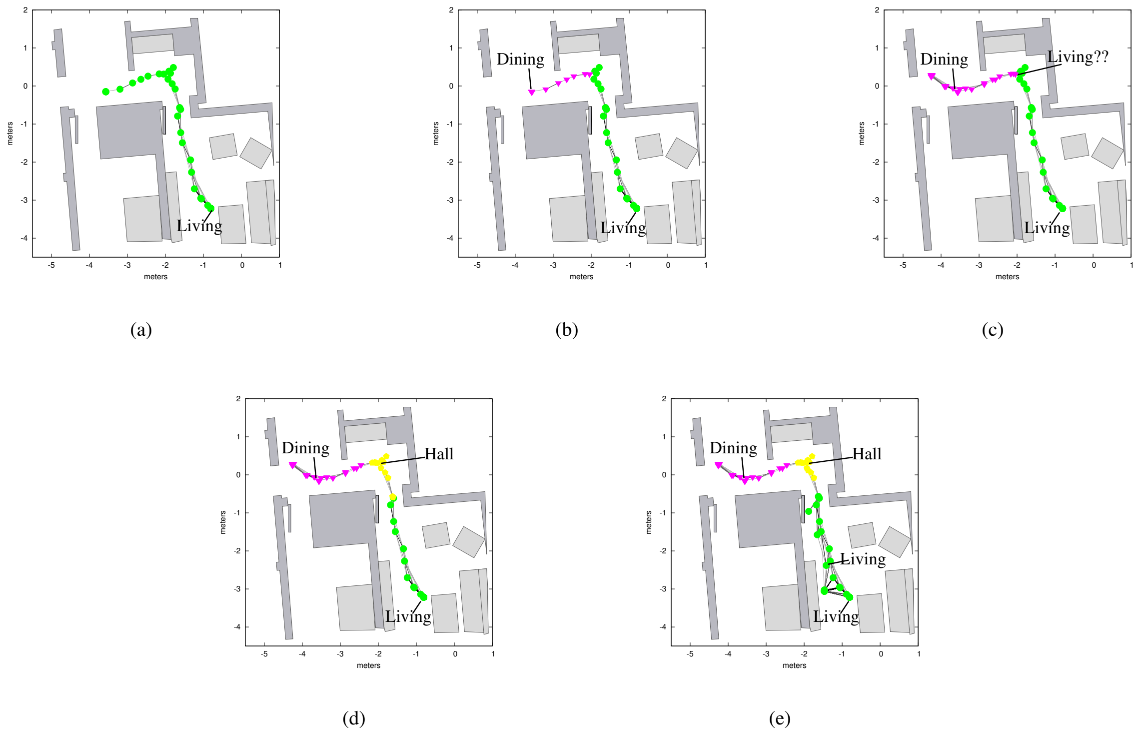


Fig. 3. The spatial representation build by the robot. The different symbols denote nodes (images) of the graph. The lines between the symbols denote links between the lines, with darker colored lines representing links with a smaller distance. Green circles denote nodes belonging to the "Living room", pink squares to the "Dining room" and yellow pentagons to the small "Hallway". Symbols linked with a label represent nodes that were labeled by the guide. In addition part of the ground-truth floor map is plotted on top for reference.

in augmented mapping systems. The robot actively asks the labels of rooms that were not labeled at the first visit and decreases the mismatch between the human representation of room transitions and the robots representation. The complete system can be run in real time on a single laptop and has been shown to work in a real home environment.

Future work is directed to gathering larger evidence for the feasibility of the interactive localization approach. The system scales well to larger environments and is flexible because it uses only a vision sensor.

REFERENCES

[1] G. Taylor A. Diosi and L. Kleeman. Interactive slam using laser and advanced sonar. In *2005 IEEE International Conf. on Robotics and Automation, ICRA 2005*, Barcelona, Spain, April 2005.

[2] Thorsten Spexard, Shuyin Li, Britta Wrede, Jannik Fritsch, Gerhard Sagerer, Olaf Booij, Zoran Zivkovic, Bas Terwijn, and Ben Kröse. Biron, where are you? - enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, October 2006.

[3] A. Saffiotti and K. LeBlanc. Active perceptual anchoring of robot behavior in a dynamic environment. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 3796–3802, San Francisco, CA, 2000. Online at http://www.aass.oru.se/˜asaffio/.

[4] E.A. Topp, H. Huettenrauch, H.I. Christensen, and K. Severinson Eklundh. Bringing together human and robotic environment representations - a pilot study. In *In Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[5] Hendrik Zender, Oscar Martinez Mozos, Patric Jensfelt, Geert-Jan M. Kruijff, and Wolfram Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6):493–502, June 2008.

[6] O. Martínez-Mozos, R. Triebela, P. Jensfeltb, A. Rottmanna, and W. Burgarda. Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems*, 55(5):391–402, 2007.

[7] J. Fritsch, M. Kleinehagenbrock, A. Haasch, S. Wrede, and G. Sagerer. A flexible infrastructure for the development of a robot companion with extensible HRI-capabilities. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 3419–3425, Barcelona, Spain, April 2005.

[8] O. Booij, Z. Zivkovic, and B. Kröse. Sparse appearance based modeling for robot localization. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1510–1515. IEEE, October 2006.

[9] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, December 1959.

Active Learning for Socially Assistive Robotics for Stroke Rehabilitation and Dementia Care

Adriana Tapus
Department of Computer Science
University of Southern California
Los Angeles, USA
Email: adriana.tapus@ieee.org

Maja Matarić
Department of Computer Science
University of Southern California
Los Angeles, USA
Email: mataric@usc.edu

***Abstract*— This paper describes an interdisciplinary research project aimed at developing and evaluating effective and user-friendly non-contact robot-assisted therapy, aimed at in-home use. Specifically, the research develops and evaluates a method of on-line user adaptation aimed at both personalizing the therapy process and maximizing its health-related outcomes. Our approach is original and promising in that it combines several ingredients that individually have been shown to be important for long-term efficacy in motor rehabilitation and cognitive skills improvement: (1) intensity of task specific training; (2) engagement and self-management of goal-directed actions. These principles motivate and guide the strategies used to develop novel user activity sensing and provide the rationale for development of socially assistive robotics therapy for monitoring and coaching users toward customized and optimal rehabilitation and care programs.**

I. Introduction

Robotic systems are now capable of social interaction with human users, presenting a new opportunity for providing individualized care. Mounting evidence shows that human users respond more readily to robots than to disembodied alternatives such as computer screens, personal digital assistants, and smart phones.

As the elderly population continues to grow, a great deal of attention and research is dedicated to assistive systems aimed at promoting ageing-in-place, facilitating living independently in ones own home as long as possible.

Two of the main problems encountered in the elder population are the Alzheimer's disease, which is a form of dementia, and stroke. In a recent report published by the American Alzheimer's Association [1], it is stated that since the incidence and prevalence of Alzheimer's disease increase with advancing age, the number of persons with the disease is expected to grow as a proportion of this larger older population. Therefore, the rapidly increasing number of people suffering from Alzheimer's disease could cripple healthcare services in the next few decades. The latest estimate is that 26.6 million people were suffering from Alzheimer's disease worldwide in 2006, and that the number will increase to 100 million by 2050, 1 in 85 of the total population. The statistics also show that stroke [2] is also a very dominant health problem with more than 15 million people suffering a stroke worldwide each year.

These individuals are high users of health care, residential care and home and community services and they need long-term care services; for example stroke survivors need to re-learn skills that were lost when part of the brain was damaged, and the intensive post-stroke rehabilitation therapy (usually around 6 hours per day) during the critical months of the post-stroke period is crucial in the recovery; also for the individuals suffering of cognitive impairment such as Alzheimer's disease, even if there is no cure, medication and special therapy can improve disease symptoms. Non pharmacological treatments focus on physical, emotional and also mental activation. Engagement in activities is one of the key elements of good dementia care. Activities (e.g., music therapy, arts and crafts) help individuals with dementia and cognitive impairment maintain their functional abilities and can enhance their quality of life. Also cognitive rehabilitation therapies that focus on recovering and/or maintaining cognitive abilities such as memory, orientation, and communication skills are other specific therapeutic protocols designed for individuals with dementia. Finally, physical rehabilitation therapies that focus on motor activities help individuals with dementia rehabilitate damaged functions or maintain their current motor abilities so as to keep the greater possible extent of autonomy.

Therefore, in this work we investigate the role of robots active learning in the assistive therapy process and we try to address the following research question: *How should the behavior and encouragement of the therapist robot adapt as a function of the users personality, preferences, physical and cognitive impairment, and task performance?*

II. Learning Methodology

Learning to adapt our daily behavior as a function of different internal and external factors it's a fundamental characteristic of humans. Creating robots capable of exhibiting similar sophisticated capabilities has proven to be a very difficult task. Therefore, providing an engaging and motivating customized protocol that is adaptable to user personality, preferences, physical and cognitive impairment, and task performance is a challenge in robotics, especially when working with vulnerable user populations, where a careful consideration of the users needs and disabilities is required.

To the best of our knowledge, no work has yet tackled the issue of robot personality and behavior adaptation as a function of user personality and task performance in the assistive human-robot interaction context. In the work described here, we address these issues and propose a reinforcement-learning-based approach to robot behavior adaptation. In the learning approach, the robot incrementally adapts its behavior and thus its expressed personality, attempting to maximize the task performance. The robot's behavior (and therefore personality and empathy) is expressed through multi-modal cues which include: interpersonal distances/proxemics, verbal, para-verbal, and non-verbal communication, and activity that will allow the robot to be responsive both in terms of temporal and social appropriateness.

We formulated the problem as policy gradient reinforcement learning (PGRL) and developed a learning algorithm that consists of the following steps: (a) parametrization of the behavior; (b) approximation of the gradient of the reward function in the parameter space; and (c) movement towards a local optimum. The main goal of our robot behavior adaptation system is to enable us to optimize on the fly the three main interaction parameters (interaction distance/proxemics, speed, and verbal and paraverbal cues) that define the behavior (and thus personality and empathy) of the therapist robot, so as to adapt it to the users profile and thus improve the users task performance. More details can be found in [3].

As a function of the user population and therefore the designed task, task performance is measured either as the number of exercises performed in a given period of time (in the post-stroke physical rehabilitation setup), or as the reaction time and the amount of vocalization (in the dementia cognitive therapy setup). Hence, the learning system changes the robot's personality, expressed through the robot's behavior, in an attempt to maximize the task performance metric.

III. Experimental Design

A. Post-Stroke Physical Rehabilitation

Two different experiments were designed in order to test the adaptability of the robot's behavior to the participants personality and preferences. The experimental task was a common object transfer task used in post-stroke rehabilitation and consisted of moving sticks from one box on the left side of the participant to another box on his/her right side. One of the boxes was on an electronic scale in order to measure the user's task performance. The task was open-ended. The subject pool consisted of 12 participants (7 male and 5 female). In order to determine the users' personality (based on the Eysenck Personality Inventory (EPI) [4] and preferences related to the therapy styles or robots vocal cues, interaction distances, and robots speed from the values used in the experiments, the participants were asked to complete a pre- and post- experiment questionnaire. The learning algorithm was initialized with parameter values that were in the vicinity of what was thought to be acceptable for both extroverted and introverted individuals, based on one of our previous study [5]

The first experiment was designed to test the robot behavior adaptation to user personality-based therapy style. The therapy styles ranged from coach-like therapy to encouragement-based therapy for extroverted personality types and from supportive therapy to nurturing therapy for introverted personality types. The vocal content for each of these scenarios was selected in concordance with encouragement language used by professional rehabilitation therapists.

It is well known that people are more influenced by certain voices and accents than others. The main goal of our second experiment was to test and validate the adaptation capability of the robot to the user preferences related to English accent and voice gender.

B. Dementia and Alzheimer's Disease Care

We designed a new experiment to improve the participants attention and consists of a cognitive game called song discovery or name that tune (i.e., find the correct button for the song, press it, and say the name of the song). The criteria for participation (in addition to the dementia diagnosis) in this experiment include the ability to read large prints and to press a button. The objective measure of this study is the reaction time for both song detection and silence detection verbally and with buttons. The main goal is to minimize the reaction time and maximize verbalization, which signifies improvement of cognitve attention.

The participants performance during the game is assessed using both data obtained from the interaction with the robot and button recordings, and data obtained from video recordings. Music therapist feedback will be gauged through a questionnaire completed at the end of the experiment. Outcomes will be quantified by evaluating task performance and time on task.

IV. Experimental Results

A. Post-Stroke Physical Rehabilitation

The pilot experimental results provided first evidence for the effectiveness of robot behavior adaptation to user personality and performance: users (control group - individuals who were not stroke patients) tended to perform more or longer trials under the personality matched and therapy style matched conditions. The result is a novel stroke rehabilitation tool that provides individualized and appropriately challenging/nurturing therapy style that measurably improves user task performance.

B. Dementia and Alzheimers Disease Care

Two focus groups were conducted at our partners sites: Silverado Senior Living and The Jewish Home Los Angeles. The preliminary focus groups and early studies already show promise for our approach. More experimental results validating our hypotheses will be available by the time of the workshop, as this paper reports on ongoing work in progress.

V. CONCLUSION

This paper presents a novel incremental learning methodology for assistive purposes. Our non-contact therapist robot monitors, assists, encourages, and socially interacts with post-stroke users and people suffering from cognitive impairement and/or dementia during rehabilitation/maintenance therapy. The experimental results provide first evidence for the effectiveness of robot behavior adaptation to user profile and performance.

ACKNOWLEDGMENT

This work was supported by the Okawa Foundation, the National Science Foundation Grant 0709296, and the USC NIH Aging and Disability Resource Center (ADRC) pilot program.

REFERENCES

[1] American Alzheimer Association *About Alzheimer's Disease Statistics*, 2007.

[2] National Institute of Neurological Disorders and Stroke. *Post-Stroke Rehabilitation Fact Sheet*, January 2006.

[3] A. Tapus, C. Tapus and M. J. Matarić. User personality matching with hands-off robot for post-stroke rehabilitation therapy. *Intelligent Service Rootics: Specia Issue on Multidicsiplinary Collaboration for Socially Assistive Robotics*, 1(2):169–183, 2008.

[4] H. J. Eysenck. Dimensions of personality: 16, 5 or 3? criteria for a taxonomic paradigm. *Personality and Individual Differences*, 12:773–790, 1991.

[5] A. Tapus and M. J. Matarić. User personality matching with hands-off robot for post-stroke rehabilitation therapy. In *Proc. International Symposium on Experimental Robotics (ISER'06)*, Rio de Janeiro, Brazil, July 2006.

Learning of Arm Exercise Behaviors: Assistive Therapy based on Therapist-Patient Observation

Ayanna M. Howard, Sekou Remy, Hae Won Park
Human-Automation Systems (HumAnS) Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA, USA

Abstract- **Machine learning techniques have currently been deployed in a number of real-world application areas – from casino surveillance to fingerprint matching. That fact, coupled with advances in computer vision and human-computer interfaces, positions systems that can learn from human observation at the point where they can realistically and reliably be deployed as functional components in autonomous control systems. Healthcare applications though pose a unique challenge in that, although autonomous capability might be available, it might not be desired. And yet, based on recent studies focused on assessment of the changing demographics of the world, there is a need for technology that can deal with the shortcomings envisioned in the workforce. Traditional roles for robotics have focused on repetitive, hazardous or dull tasks. If we take the same stance on healthcare applications, we find that some therapeutic activities fall under this traditional classification due to the long-repetitive nature of the therapist-patient interaction. As such, in this paper, we discuss techniques that can be used to model exercise behavior by observing the patient during therapist-patient interaction. The ultimate goal is to monitor patient performance on repetitive exercises, possibly over the course of multiple days between therapy sessions.**

I. INTRODUCTION

Physical therapy is a very practitioner intensive process. When patients enter into the process they are often required/asked to perform exercises that they have been shown how to do when they are at home between visits. Proper compliance is strongly correlated with shorter time to recovery as well as reduction of pain in the long term [1]. During the time between therapy sessions there are many factors which affect patient compliance, including forgetfulness, lack of motivation, boredom, and lack of instant feedback. To deal with these issues, researchers have shown the positive use of robots in assistive therapy applications ranging from stroke rehabilitation [2] to motor development in children [3].

In many of these applications, if we can correctly identify and match patient exercise behavior based on characteristics learned during previous therapist-patient session, we can develop a monitoring mechanism to provide feedback for patient recovery. To enable this capability, we present two methods that utilize image-based observation as a means of gathering sensing information, and classification to identify subsequent patient behavior based on observations during the therapist-patient session.

II. ALGORITHM: LEARNING EXERCISE BEHAVIORS

A. *Learning of Exercise Primitives through Observation*

Learning of exercise primitives involves modeling an exercise scenario by sequencing a series of repetitive motion behaviors together. A motion behavior is used to represent an interpretation of the basic movements of an arm exercise. It is not designed to compute specific motion vectors (such as specific arm joint trajectories), but rather to provide information about general movements. We define a motion vector

$$\mathrm{M}_v = (d, v) \tag{1}$$

where d represents the direction of motion and v represents the velocity of motion. In addition, the possible values associated with d and v are discretized based on pre-defined linguistic classes, as depicted in Table I. As such, there is a finite number of motion vectors that exist for defining a low-level motion behavior. We define this finite set of possible motion vectors as the motion class κ_{motion}.

Table I. Motion behavior definition structure

Motion Parameter	Linguistic Values
Direction (d)	Left, Right, Up, Down
Velocity (v)	Slow, Fast

The direction parameter represents the absolute direction of a hand with respect to a world coordinate system. The following direction vectors are used to classify this motion parameter:

$$LEFT = \begin{bmatrix}-1\\0\end{bmatrix}, RIGHT = \begin{bmatrix}1\\0\end{bmatrix}, UP = \begin{bmatrix}0\\-1\end{bmatrix}, DOWN = \begin{bmatrix}0\\1\end{bmatrix}$$

The velocity of the motion behavior, v, is measured as follows:

$$v = \frac{\Delta p}{\Delta t} \text{ (px/s)} \tag{2}$$

Δp is defined, with respect to an observation, as the distance between the location of the hand when a motion initiates and terminates. Δt is measured by counting the frame numbers during a motion and dividing it by the average frame rate of the camera. Since the velocity required in this study need not be precise, it is reclassified as a speed: SLOW/FAST. If a motion is faster than the overall sequence speed average, it is defined as FAST, and as SLOW otherwise. As an illustrative example, Table III shows the association between low-level motion behaviors and the resulting motion vectors.

Table II. Association - motion behaviors and vectors

Illustrative Description of Motion Behavior	Motion Vector
Human quickly lifts hand up	(Up, Fast)
Human shakes hand to the right	(Right, Fast)

The goal of the motion behavior analysis process is to populate instances of the motion vector based on observation of a human exercise action (such as depicted in Figure 1). This process is executed by computing a motion gradient during human exercise and fitting the motion gradient to the pre-defined motion class. The motion behavior analysis process is further described in [4]. Once motion behaviors are identified, the sequence of motion behaviors associated with an exercise scenario are stored and labeled (by the therapist). After therapist-patient interaction, the system is to match the stored therapy exercise information to the patient during subsequent exercises using the same motion behavior analysis process.

Figure 1. Sequence of images captured during observation (top) 180° left shoulder abduction (middle) 90° left shoulder abduction (bottom) right shoulder rotation

B. *Learning Exercise Behaviors through Observation*

In the previous approach, image-based methods were used to construct an exercise scenario from a sequence of identified motion behaviors. In the next approach, we utilize a method that classifies the entire exercise scenario using a single representation. Based on imaging the patient during a therapy session, a texture based feature vector is first generated for each image (frame) and stored in a database. This database is then used to train an adaptive classifier to classify the elements in the dataset, using the approach as described in [5]. During subsequent exercise, the method presented in [6] is used to extract period and frequency information for the captured data in order to generate a mapping between observed state and its position in the exercise cycle. In this step, we assume only one exercise is exhibited in the captured data sequence. After therapist-patient interaction, a measure of similarity using the 2D Kolomogorov Smirnov test [7] is calculated to determine the statistical goodness of fit between pairs of exercise behaviors. This test is used to determine which of the stored therapy exercises the patient is performing during subsequent exercises.

III. Interaction between User and Robot

In the subsequent section, we outlined two complimentary methods to correctly identify and match patient exercise behavior with information captured during therapist-patient interaction. Since exercise motions depend on individual capability (and can vary both between individual subjects as well as between the same subject during different exercise scenarios), the role of the therapist during these scenarios is 1) to correctly position the robot such that important body features are in view of the robot, and 2) to correct the labeling of the behaviors during subsequent sessions with the patient. In theory, to allow for development of a monitoring mechanism to provide feedback for patient recovery, the therapist must interactively work with the robot to correct learned knowledge.

IV. Experimental Setup

To generate data akin to that expected with a therapy patient, the exercises, as shown in Table III, were first performed during a simulated therapist-patient session, and then subsequently, in random order, repeated with varying rates of execution (Figure 1).

Table III. Exercise Cases Considered

Shoulder Abduction Seated (right, left, 90°, 180°)
Shoulder Rotation Seated (right, left)
Shoulder Abduction Standing (right, left, 90°, 180°)

The goal in implementing the two different methodologies is to assess the capability of the system to correctly identify the patient exercise and determine the system characteristics that contribute to success of each approach. Preliminary analysis show that the recognition methods can uniquely identify patient behaviors as long as the following assumptions hold: 1) there is no significant change in the activity performed during subsequent sessions, 2) the therapist correctly shows the patient how to perform the exercises safely, and the patient is able to comply, 3) the patient's appearance remains relatively consistent during subsequent sessions, and 4) the robot can position its camera as appropriate to capture the execution of each exercise.

V. CONCLUSIONS

In this paper, we present two methods that enable learning of therapy exercises performed during a therapist-patient session. The approach uses vision as a means of observing the user during task execution. The stored exercise sequence can then be utilized by the system to match subsequent patient behavior. Future work involves developing approaches to extract specific performance metrics (i.e. speed and frequency) to provide feedback to the therapist for enhancing patient recovery.

ACKNOWLEDGEMENT

This research is based upon work supported by the National Science Foundation under Grant No. IIS-0705130.

REFERENCES

[1] E.M. Wakiji, “Mapping the literature of physical therapy,” *Bull Med Libr Assoc.*, 85(3): 284–288, July 1997.

[2] R. Gockley, M.J. Mataric, “Encouraging Physical Therapy Compliance with a Hands-Off Mobile Robot ,” *Proc. of Human-Robot Interaction*, pp. 150-155, Utah, March 2006.

[3] J.C. Galloway, J. Rhu, and S. Agrawal, “Babies driving robots: Self-generated mobility in very young infants,” *Journal of Intelligent Service Robotics,* 2008.

[4] A. M. Howard, H.W. Park, C. Kemp, “Extracting Play Primitives for a Robot Playmate by Sequencing Low-Level Motion Behaviors,” IEEE Ro-Man, Germany, to appear 2008.

[5] S. Remy, A. Howard, “In Situ Interactive Teaching of Trustworthy Robotic Assistants,” *IEEE Int. Conf. on Systems, Man, and Cybernetics*, Montreal, Canada, Oct. 2007.

[6] W. A. Sethares and T. W. Staley, "Periodicity Transforms", *IEEE Trans. on Signal Processing*, pp. 2953-2964, Nov 1999.

[7] S.A. Metchev, J.E. Grindlay, “A two-dimensional Kolmogorov-Smirnov test for crowded field source detection: ROSAT sources in NGC 6397,” Mon Not R Astron Soc., v335, pp. 73-83, 2002.

Learning Qualitative Models by an Autonomous Robot

Jure Žabkar and Ivan Bratko
AI Lab, Faculty of Computer
and Information Science,
University of Ljubljana,
SI-1000 Ljubljana, Slovenia
Email: jure.zabkar@fri.uni-lj.si

Ashok Mohan and Erwin Prassler
University of Applied Sciences
Bonn-Rhein-Sieg
Grantham-Allee 20,
53757 Sankt Augustin, Germany

***Abstract*— In this paper we present a qualitative exploration strategy for an autonomous robot that learns by experimentation. Particularly, we describe a domain in which a mobile robot observes a ball and learns qualitative prediction models from its actions and observation data. At all times it uses these models to predict the results of the actions that it has decided to execute and to design new experiments that would lead it to learn a better model of the world, and for planning of the execution of these experiments. The models also represent the insights of the robot's knowledge. We experimentaly evaluate the exploration strategy.**

I. Introduction

The idea of autonomous robots that are capable of learning by themselves, without any human intervention is one of the most fundamental goals of AI. Among several paradigms of learning, learning by experimentation demands no teacher, but rather learns autonomously, interacting with the real world. In this paper we present a showcase in which an autonomous robot is learning qualitative models by conducting experiments in its environment.

There are several ways of how the robot chooses its actions, designs and plans experiments. In order to learn efficiently, the strategy which it uses to explore its environment is very important. We propose a qualitative exploration strategy for autonomous robot learning. We evaluate our strategy by comparing it to random strategy. The results show that using our strategy, the robot is learning faster and it learns better models. We consider learning of *qualitative* models an important aspect. This is due to the fact that qualitative models are easier to learn and sufficient to design and plan the experiment. They reduce the complexity of numerical models considerably and also enable humans to easily understand what the robot has learned.

The robot has no prior knowledge about its environment. In particular, it has no knowledge regarding the relations between its actions and observations. Its task is collecting the data and gradually learning a model which it immediately uses for moving and designing new experiments. Its goal is to learn a model that would relate its actions to its observations. At each step, the robot decides on one of several possible actions. It then uses its current model to predict the result of its action, executes the action and collects observations. It then compares its prediction with the actual observations. The result of this comparison leads to further experiments that helps to revise the model.

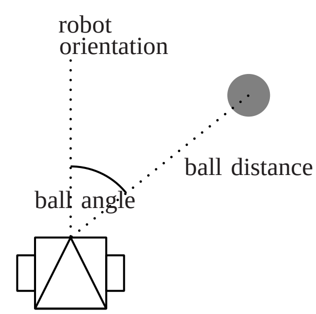


Fig. 1. The robot and the ball.

Regarding related work, our approach is most similar to [1]. However, one of our general goals is to obtain the insights of robot's knowledge, so the models should be understandable to humans rather than black boxes. In this respect, our work is similar to [2]. A broader context also includes [3, 4, 5, 6].

II. Learning qualitative models by experimentation

Our problem domain consists of a mobile robot and a ball. The robot observes its distance to the ball (ball distance, denoted by bd) and the angle between its orientation and the ball (ball angle, denoted by ba), as shown in Fig. 1.

The robot is of differential drive type (Khepera-like) and moves by setting the speeds of the left and the right wheel (L and R respectively). In our case, L and R are always positive, and the robot was restricted to choose between speeds 4 and 5 only. So the robot can move straight ahead ($L = R = 5$), right ($L = 5, R = 4$) and left ($L = 4, R = 5$), as shown in Fig. 2. The robot is not aware of any coordinate system. It is only aware of the actions it performs (L and R) and the observations from the sensors (bd and ba).

The overall goal that we want the robot to achieve is that

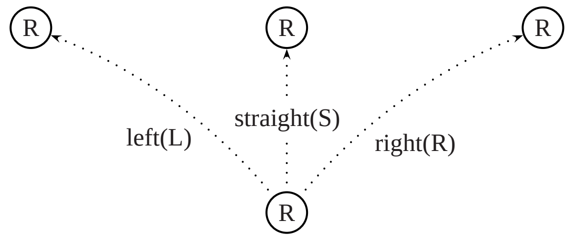


Fig. 2. The actions of the robot.

it learns a qualitative model describing the relations between its actions and observations. By densely sampling the whole space of above mentioned variables and learning a qualitative tree we obtained an "almost ideal" model of our domain. Note, that we did not use this model in any other way than to see for ourselves what the robot should eventually learn. This "almost ideal" qualitative tree for our domain is shown in Fig. 3. The notation used in the model is explained below.

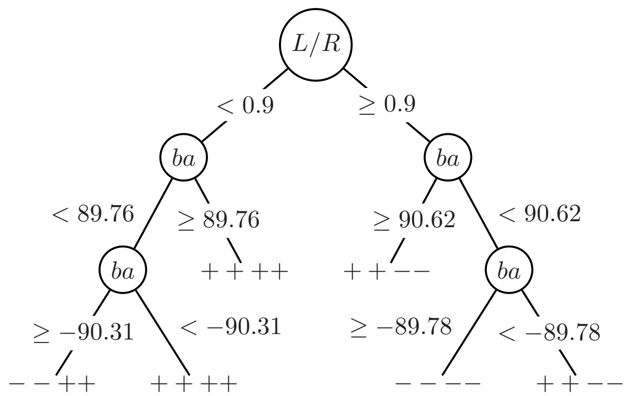


Fig. 3. The "almost ideal" model of the robot in our domain.

We have performed all the experiments in the simulator Simon which is a part of the machine learning framework Orange [7].

The robot learns its first model from a small dataset of just 10 samples collected by random movement. This initial model is not very accurate and useful. Nevertheless, it enables the robot to use it for making predictions about further actions.

The ability to make predictions enables the choice among learning strategies. A learning strategy determines the next action. The most primitive learning strategy is random strategy, in which the robot chooses one of its three possible actions at random. Random movement is thus defined by actions rather than by the robot's positions. The latter is not even possible since in our case the robot is not aware of its coordinates and can not choose to navigate in any coordinate system.

The robot is supposed to learn the relations between its actions and observations. In our simple example, it has two actions (L and R) and two observation variables (ba and bd), so it should learn $bd = Q(sS_L)$, $bd = Q(sS_R)$, $ba = Q(sS_L)$ and $ba = Q(sS_R)$, where sign s is $+$ or $-$ and $L = \dot{S}_L, R = \dot{S}_R$, where S_L and S_R are the paths of the left and the right

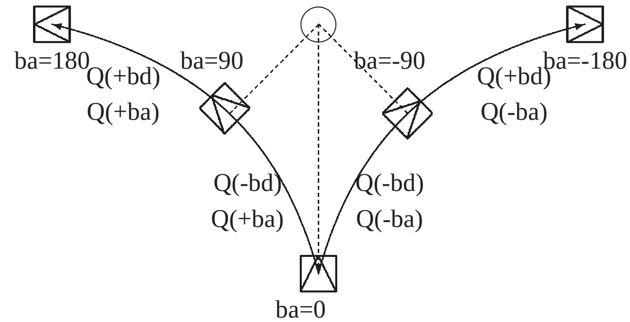


Fig. 4. The various angles when robot is turning left and right from $ba = 0$

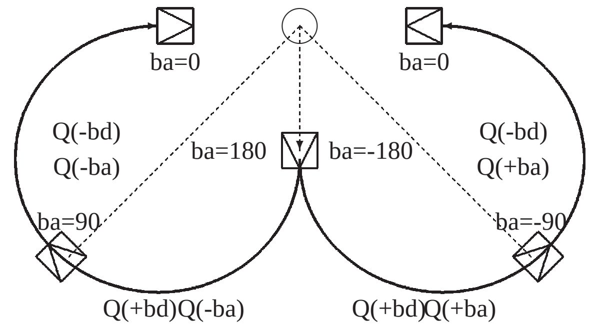


Fig. 5. The various angles when robot is turning left and right from $ba = 180$ or $ba = -180$

wheel respectively. In these equations, Q stands for qualitative relation as described in [8]. In the paper, we use a shorter notation, e.g. "$++--$", giving only the signs s in the above mentioned order. So "$++--$" means: $bd = Q(+S_L)$, $bd = Q(+S_R)$, $ba = Q(-S_L)$ and $ba = Q(-S_R)$. In words: ball distance is increasing when S_L and S_R are increasing (i.e. $L, R > 0$), and ball angle is decreasing when S_L and S_R are increasing. We define the class C of this domain as a 4-tuple of signs as just described. Figures 4 and 5 clearly shows the regions of different values of class C.

Qualitative models that the robot is learning are in the form of qualitative trees and qualitative non-deterministic finite automata (NFA). The robot uses algorithm Padé [8] with decision trees to learn qualitative trees while it builds an NFA from the temporal sequence of its actions and observations. The initial set of attributes includes L, R, ba, bd and the class C. To this set, Padé adds a newly constructed attribute L/R, obtained by the chain rule, dividing the derivatives of each wheel's path w.r.t. time. The attribute L/R describes the qualitative relation between both speeds and can, as we shall see, explain the left and right turns. Using the chain rule for attribute construction is a general principle and is not specifically added to this domain.

The robot's exploration algorithm includes three strategies that strive to guide the learning towards the final goal. The most primitive is the *random strategy* — the robot moves

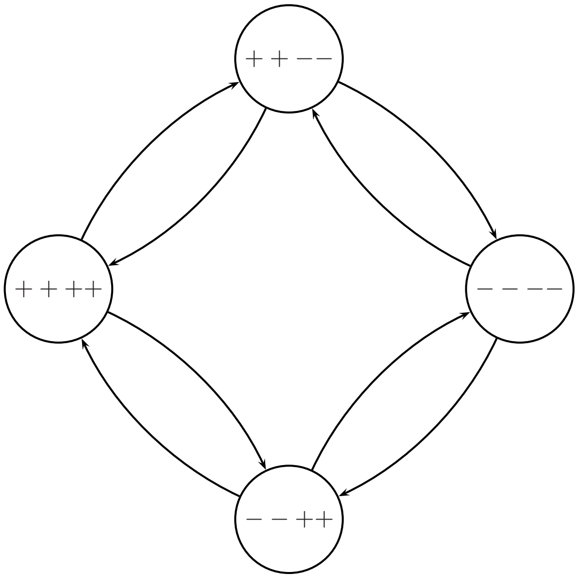


Fig. 6. The NFA learned by the robot.

randomly choosing the actions from its set of available actions. The second strategy we call *uniform strategy*; it is used when the robot wants to sample the actions so that their distribution is uniform. Uniform distribution of actions assures that the robot is not biased towards one of the actions, e.g. going straight ahead all the time. At first glance it may seem that uniform and random strategies are the same, but the difference lies in the fact that uniform strategy also accounts for the action executed using persistent strategy. The third strategy is called *persistent strategy* — the robot keeps executing the same action for some time and is collecting more learning examples of the same kind.

The robot uses random strategy only for its first ten moves when it has no knowledge about its environment and the random choice is the best it can make. After it collects the first ten learning examples it can already build a first model and start using it. At this time, it changes the strategy to *uniform* and enters the main loop in which it is updating and improving the model.

The main loop starts with choosing the next action based on the current strategy (either uniform or persistent). After the robot picks the action it uses the current qualitative tree to make the prediction using the current state and the action. When it makes the prediction it executes the action and observes the result. It compares its own prediction with the actual observation. If they match, the robot continues with persistent strategy, otherwise the robot is "surprised" and motivated for further exploration of the unknown behaviors. The reason for the mismatch is the false prediction of qualitative behavior, i.e. the signs in the class value were predicted wrongly. The robot updates the NFA with a new state and transition and also updates the qualitative tree. After it updates the model, the robot starts designing a new experiment and planning its actions so that it could carry out the designed experiment. For this purpose it maintains a frequency table of class values and it observes the difference between the number of examples in the current state of the NFA and the one with the lowest frequency in the table. If the number of examples in the current state of the NFA is greater than a threshold, it selects uniform strategy and picks persistent otherwise. This finishes one iteration of the loop and starts a new one.

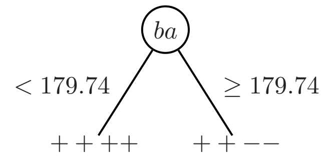


Fig. 7. The model created by the robot after 19 steps.

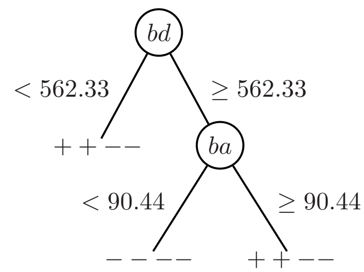


Fig. 8. The model created by the robot after 1000 steps.

III. RESULTS

The exploration algorithm from the previous section enables the robot to learn by experimentation in an efficient way. To confirm the latter, in this section we compare our approach to the pure random strategy. Again, we stress that random strategy does not mean random sampling of the coordinate space but rather choosing the actions randomly.

In random strategy we use a parameter *duration* which defines the frequency for choosing a random action. If *duration* $= 1$ the action is chosen randomly on each simulation step while for *duration* $= n$ it is chosen only each n-th step and maintained the same in between. The latter is actually not a pure random strategy but rather a mixture of random and persistent. We use it for comparison anyway since the pure random strategy performs extremely poor.

We ran 3 runs of each random strategy, varying *duration* and 9 runs with different initial positions of the robot with our exploration algorithm. We manually determined the point at which the robot learned the desired model. We measured the time it had needed to learn the model in the number of steps it performed until that state. Using a pure random strategy, the robot never managed to learn the model and the process was terminated after 30000 steps. Using our exploration strategy it always learned the model we expected in the average of 3582 steps. Table I presents the results over different runs, the averages and standard errors.

	Random		our exploration strategy	
Run	Stepsize	Steps taken to reach best model	Stepsize	Steps taken to reach best model
1		Not until 30000	1	2674
2	1	Not until 30000	1	3685
3		Not until 30000	1	1991
4		Not until 30000	1	2078
5	10	Not until 30000	1	3530
6		Not until 30000	1	3254
7		15967[a]	1	7317
8	100	Not until 30000	1	4866
9		27654[b]	1	2843

[a] Even this does not result in the ideal model, but very close to it
[b] This resulted in a model separated at the root by L instead of L/R

TABLE I

COMPARISON BETWEEN RANDOM ACTION SELECTION AND OUR EXPLORATION STRATEGY PRESENTED HERE.

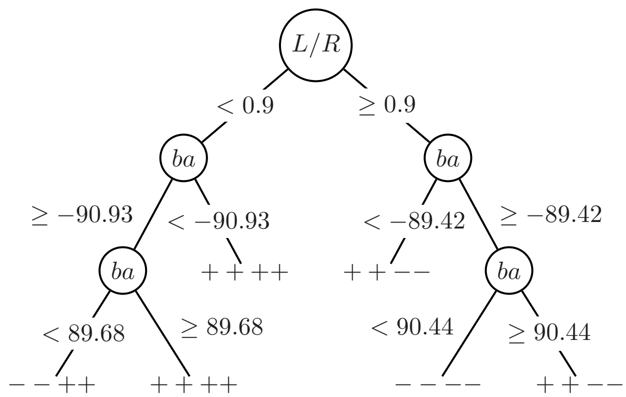


Fig. 9. The final model created by the robot after 2674 steps.

The results show that the robot learns significantly better and faster with our exploration algorithm as opposed to the pure random strategy or random-persistent strategies. We have no formal proof to explain why our strategy works. Nevertheless, it is clear from the way humans experiment that we pursue one direction until there arises a reason or motivation to change it.

IV. CONCLUSION

We showed a simple example of a robot that is capable of learning by making experiments in its environment. The exploration algorithm that we presented proved to be a useful tool for the autonomous learner that has to design, plan and execute the experiments in order to obtain some knowledge about how its actions influence its observations in the given world. One of the contributions in our opinion is the use of qualitative models only and the combination of qualitative tree and the NFA. Both models do not only suffice to support the robot in its actions, but also offer insights into the knowledge that the robot acquired in the learning process. Further, we believe that our approach can be generalized to other more complex domains and that it can scale well due to the simplicity of learning the qualitative models.

ACKNOWLEDGMENT

The work described in this article has been funded by the European Commission's Sixth Framework Programme under contract no. 029427 as part of the Specific Targeted Research Project XPERO ("Robotic Learning by Experimentation").

REFERENCES

[1] Oudeyer, P.-Y. and Kaplan, F. and Hafner, V. *I*ntrinsic Motivation Systems for Autonomous Mental Development. IEEE Transactions on Evolutionary Computation, Vol. 11(6), 2007.
[2] Modayil, J. and Kuipers, B. *W*here do actions come from? Autonomous robot learning of objects and actions. AAAI Spring Symposium Series 2007, Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems,2007
[3] Barto, A. G. and Singh, S. and Chentanez, N. *I*ntrinsically motivated learning of hierarchical collections of skills. International Conference on Developmental Learning,2004
[4] Hart, S. and Grupen, R. and Jensen, D. *A* relational representation for procedural task knowledge. Proc. 20th National Conf. on Artificial Intelligence,2005
[5] Kuipers, B.; Beeson, P.; Modayil, J.; and Provost, J. *B*ootstrap learning of foundational representations. Connection Science, 18(2),2006
[6] Stoytchev, A. *B*ehavior-grounded representation of tool affordances IEEE International Conference on Robotics and Automation (ICRA),2005
[7] Zupan, B. and Leban, G. and Demar, J. *O*range: Widgets and visual programming, A White Paper Pad 2004
[8] Žabkar, J. and Bratko, I. and Demšar, J. *L*earning Qualitative Models through Partial Derivatives by Pad Proceedings of the 21th International Workshop on Qualitative Reasoning,2007