Proceedings 2005 IEEE International Symposium on
Computational Intelligence in Robotics and Automation
June 27-30, 2005, Espoo, Finland

An Approach for Programming Robots by Demonstration: Generalization Across Different Initial Configurations of Manipulated Objects *

Aris Alissandrakis, Chrystopher L. Nehaniv, Kerstin Dautenhahn and Joe Saunders
Adaptive Systems Research Group
University of Hertfordshire, School of Computer Science
College Lane, Hatfield, Hertfordshire, AL10 9AB, UK
a.alissandrakis@herts.ac.uk

***Abstract*— Imitation is a powerful learning tool that can be used by a robotic agent to socially learn new skills and tasks. One of the fundamental problems in imitation is the *correspondence problem*, how to map between the actions, states and effects of the model and imitator agents, when the embodiment of the agents is dissimilar. In our approach, the matching depends on different metrics and granularity. Focusing on object manipulation and arrangement demonstrated by a human, this paper presents JABBERWOCKY, a system that uses different metrics and granularity to produce action command sequences that when executed by an imitating agent can achieve corresponding effects (manipulandum absolute/relative position, displacement, rotation and orientation).**

Based on a single demonstration of an object manipulation task by a human and using a combination of *effect metrics*, the system is shown to produce correspondence solutions that are then performed by an imitating agent, generalizing with respect to different initial object positions and orientations in the imitator's workspace. Depending on the particular metrics and granularity used, the corresponding effects will differ (shown in examples), making the appropriate choice of metrics and granularity depend on the task and context.

***Index Terms*— Imitation and social learning, correspondence problem, space of effect metrics, human-robot interaction, programming by demonstration.**

I. Introduction

Imitation is a powerful learning tool when social interaction between human and robots is required. Robotics researchers are inspired from imitation and social learning in animals and humans to create controllers for their autonomous robots, using suitable behaviours for adaptive learning [1]–[6]. Having a robot observe and learn to perform a task from an experienced teacher presents a more flexible and adaptive solution than explicit pre-programming and restrictive hardwiring. The learning process can be faster as no direct programming is required. The expert should just by performing and thus demonstrating the task pass the required knowledge to the robot, which in turn may be used as a model to be imitated by other robots.

The work presented in this paper is motivated by the EU Integrated Project COGNIRON ("The Cognitive Robot Companion") and addresses the problem of how to teach a robot new motor skills and complex tasks through human demonstration. The learning algorithms to be developed should be general and address fundamental questions of imitation learning, applied to manipulation tasks. For example a robotic companion at home could acquire knowledge of e.g. arranging some household objects on a table from observing its human owner. Acquiring such skills socially requires matching different aspects of the effects that the human actions have on objects in the environment. Also the various contexts within which a task is replicated might require its generalization to various settings (see examples below) and to other types and shapes of manipulated objects.

In order to study the most interesting and significant problems, a general agent-based perspective can be used, considering imitation as the behaviour of an autonomous agent in relation to its environment, including other autonomous agents. The mechanisms underlying imitation are not separated from the behaviour-in-context, including the social and non-social environments, motivations, relationships among the agents, their embodiments, the agent's individual and learning history, etc. [7]. Such a perspective can help to unfold the full potential of research on imitation and help in identifying challenging and important research issues, including five central questions that an imitating agent must address: *who* to imitate, *when* to imitate, *what* to imitate, *how to* imitate and how to *evaluate* a successful imitation. Each question presents its own difficulties and research problems. An integrated approach to these questions must be the ultimate goal of work on imitation in adaptive systems. The work presented in this paper concentrates on the *how to* problem.

A. Correspondence Problem

A fundamental problem when learning how to imitate is to create an appropriate (partial) mapping between the actions afforded by particular embodiments to achieve corresponding states and effects by the model and imitator agents (solving a *correspondence problem*) [8]. For similar embodiments, this seems to be straightforward (although

*The work described in this paper was conducted within the EU Integrated Project COGNIRON ("The Cognitive Robot Companion") and was funded by the European Commission Division FP6-IST Future and Emerging Technologies under Contract FP6-002020.

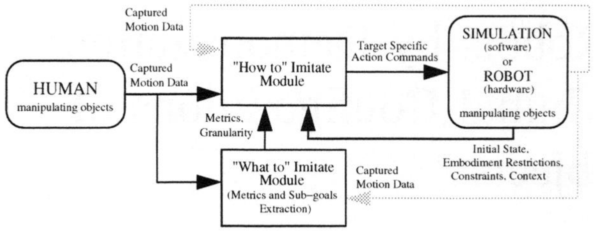

Fig. 1. **The JABBERWOCKY system architecture.** Using data captured from a human and given appropriate metrics and sub-goal granularity, the multi-target system can produce action command sequences that when executed by a software or hardware agent can achieve corresponding actions, states and/or effects. The corresponding actions, states and effects as demonstrated by the imitator can also be captured and used as a demonstration for another imitating agent (grey arrows). Differently embodied and constrained target systems in various contexts need to be supported.

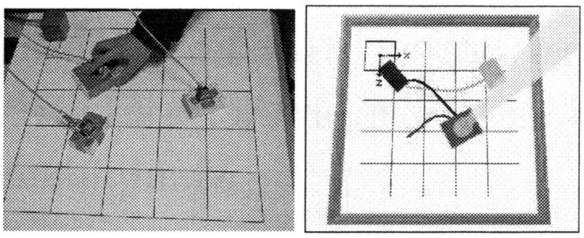

Fig. 2. **A picture of the experimental set-up for the demonstrator (left) and a screenshot of the imitator platform (right).** On the left, the human demonstrator manipulates (moves and rotates) three objects inside the workspace grid. On the right, the imitator platform (simulated in Webots) is shown, with the robotic manipulator visualized as a vertical cylinder that can translate and rotate objects. The objects in the simulation leave a coloured trail as they move, to aid the visualization of their trajectories.

it actually involves deep issues of perception and motor control). But once the assumption that the agents belong to the same 'species', i.e. have sufficiently similar bodies and an equivalent set of actions, is dropped, as with a robot imitating a human, the problem becomes more difficult and complex. Even among biological agents, individual differences in issues of perception, anatomy, neurophysiology, and ontogeny can create effectively dissimilar embodiments between members of the same species. A close inspection of seemingly similar artificial agent embodiments can yield similar conclusions due to issues like individual sensor and actuator differences (hardware) or the particular representations and processing that these agents employ (software). In our setting, it will be desirable to have different kinds of agents in the learning process, i.e. humans and robots interacting socially.

The following statement of the *correspondence problem* [9]–[11] draws attention to the fact that the model and imitator agents may not necessary share the same morphology or may not have the same affordances:

> Given an observed behaviour of the model, which from a given starting state leads the model through a sequence (or hierarchy [or program]) of *sub-goals* in *states*, *action* and/or *effects*, one must find and execute a sequence of actions using one's own (possibly dissimilar) embodiment, which from a corresponding starting state, leads through corresponding sub-goals - in corresponding states, actions, and/or effects, while possibly responding to corresponding events.

In this approach, an imitator can map observed actions of the model agent to its own repertoire of actions using the correspondence found by solving the correspondence problem, as constrained by its own embodiment and by context [9]–[11]. Qualitatively different kinds of social learning result from matching different combinations of matching actions, states and/or effects at different levels of granularity [12].

Artificial agents that have the ability to imitate may use (perhaps more than one) metric to compare the imitator agent's own actions, states and effects with the model's actions, states and effects, in order to evaluate the imitation attempts and discover corresponding actions that they can perform to achieve a similar behaviour. The choice of metrics used is therefore very important as it will have an impact on the quality and character of the imitation. Many interesting and important aspects of the model behaviour need to be considered, as the metrics capture the notion of the salient differences between performed and desired actions and also the difference between attained and desired states and effects [10], [11]. The choice of metric determines, in part, *what* will be imitated, whereas solving the correspondence problem concerns *how* to imitate [7]. In general, aspects of action, state and effect as well as the level of granularity (*what to imitate*) do all play roles in the choice of metric for solving the problem of *how to imitate* [10], [13], [14]. On-going research is thus addressing the complementary problem of how to *extract sub-goals* and *derive suitable metrics* automatically from observation [10], [12], [14], [15].

II. THE JABBERWOCKY SYSTEM

In previous work we have developed ALICE (**A**ction **L**earning via **I**mitating **C**orresponding **E**mbodiments), a generic framework for solving the correspondence problem [13], [16]. The ALICE framework builds up a library of actions from the repertoire of an imitator agent that can be executed to achieve corresponding actions, states and/or effects to those of a model agent (according to given metrics and granularity).

The ALICE framework provides a functional architecture that informs the design of robotic systems that can learn socially from a human demonstrator. For the COGNIRON project we are currently developing JABBERWOCKY, a system that uses captured data from a human demonstrator to generate appropriate action commands (see Figure 1). The action commands can be targeted for various software and hardware platforms. These actions will allow the imitating agent to achieve corresponding actions, states and/or effects, depending on the given (relevant to the demonstrated task and context) metrics and granularity (provided by a *what to imitate* and *sub-goal extraction* module), embodiment restrictions and constrains (imposed

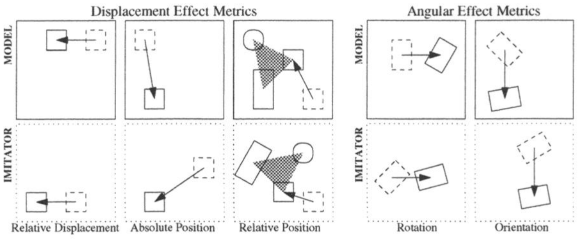

Fig. 3. **A selection of *displacement* (left) and *angular* metrics (right).** To evaluate the similarity between object displacements, the *relative displacement*, *absolute position* and *relative position* effect metrics can be used. To evaluate the similarity between object rotations, the *rotation* and *orientation* effect metrics can be used. The second row shows the way the corresponding object (in a different workspace) needs to be moved or rotated by an imitator to match the corresponding effects. The grey triangles are superimposed to show that for the *relative position* effect metric, the relative final positions of the objects are the same.

by the targeted imitator platform), and possibly different initial state of the objects in the environment.

The system bears some similarity to the one presented by Kuniyoshi et al. in [6], but with the main differences being that it is more flexible in that it can use any given metric and granularity and that it is designed to be able to generate action commands targeted for a variety of target platforms, both in software and hardware to match different aspects and achieve various types of social learning. The multiplatform targetability of JABBERWOCKY is shown in [17]. Here we assess its generalization capabilities across dissimilar initial object configurations, for different metrics.

A. Matching Human Manipulation

In the work described in this paper, the demonstrated task consists of manipulation of three coloured block objects moved and rotated on a 2D workspace surface by a human acting as the demonstrator (see Figure 2, left). Using the Polhemus LIBERTY^{TM} motion capture system, a sensor is attached on top of each object, giving the position of the object's center and also the object's orientation, relative to one of the workspace corners (each frame sampled every 15 - 20 msec). The experiments described in this paper focus on object manipulation and arrangement, so only the *effects* (the position and the orientation of the objects) are captured, omitting the demonstrator's actions (arm movements) and states (body posture). In ongoing work, three (or more) additional sensors can be used, one attached to the human torso and one at each hand, providing additional information about the demonstrator's actions and states.

For the *how to imitate* module we developed a simulation of the 2D workspace that can handle various 'block' objects moving and rotating around, accounting for object collisions and workspace confines. This simulation can replay the captured demonstration data at a given granularity, displaying the trajectory and orientation of the objects as they move and rotate on the workspace, from the initial configuration to the current frame. Starting from a different initial configuration of the same (or different) corresponding objects on a different workspace, the simulation can plan their displacement and rotation according to a combination of given *effect* metrics (described below). But some displacements or rotations, although minimizing the given metric, might be invalid if the path or final position is occupied by other objects. The *how to imitate* module will then have to discover an alternative path to achieve the same effects according to the metric. To imitate and achieve similar effects as the model, a target imitator then moves and rotates the objects according a generated sequence of command instructions that take into account constraints on the imitator's embodiment. Several target imitator platforms are supported.

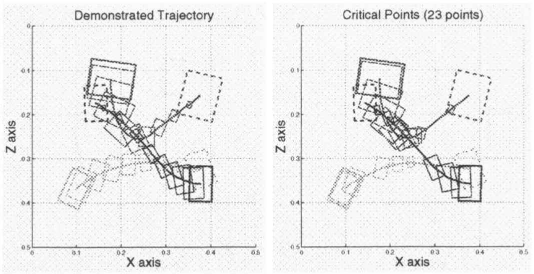

Fig. 4. **An example of a captured human demonstration (left), and the extracted critical points (right).** The colours (red, green and blue) indicate the three different objects. The dotted outlines indicate the initial position and orientation of the objects, while the solid thick outline the final position. For the demonstration data, the intermediate object's position and orientation is shown with solid thin outlines, linearly scaled (at time intervals equal to one tenth of the overall trajectory only, for clarity) to indicate the direction of the movement. For the critical points, each object's position and orientation is shown at every critical point, again linearly scaled.

The imitator platform used for the experiments presented in this paper was implemented using the Webots^{TM} robot simulation software. The imitator's workspace contains three objects, of the same size and color as the corresponding objects in the demonstrator's workspace (in this case). The imitator is embodied as a single arm manipulator, positioned above the workspace and able to pick-up, move and rotate each of the three objects (see Figure 2, right). This embodiment, although dissimilar to the one of the human demonstrator, is nevertheless able to match both *displacement* and *angular* effect aspects of the demonstration. As the objects are moved (and rotated) around the workspace by the manipulator in the simulation, they leave behind a coloured trail (of same color as themselves) to help visualize the imitated trajectories. The manipulator is shown as a vertical yellow cylinder mounted at the end of a bar positioned above the workspace, and begins and ends its trajectory from a fixed resting location.

B. Metrics and Granularity

Towards a characterization of the *space of effect metrics*, we explored absolute/relative angle and displacement aspects and focused on overall arrangement and trajectory of manipulated objects. Focusing on aspects of orientation and displacement of the manipulated objects, two types of effect metrics can be used. The first type evaluates an

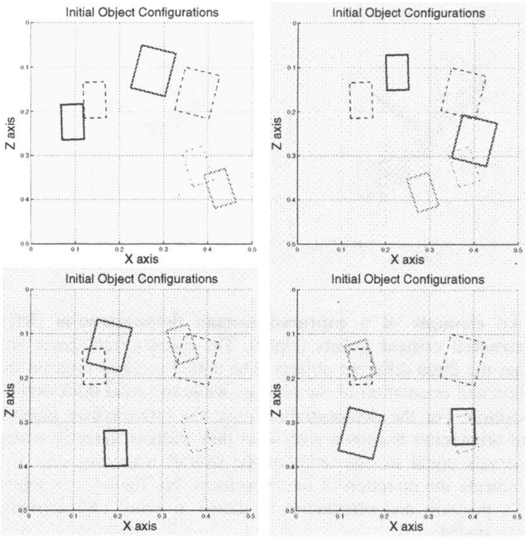

Fig. 5. **Four examples of dissimilar initial object positions (the object orientations remain the same).** The dotted outlines indicate the initial position and orientation of the objects in the demonstrator's workspace (the captured demonstrated task is shown in Figure 4, left) and the solid outlines the (dissimilar) initial configuration of the objects in the imitator's workspace. In the top right the object's relative positions are rotated by 45° (clockwise) relative to the center of the workspace, in the lower left by 90° (anti-clockwise) and in the lower right by 180°.

object's movement and position on the workspace (*relative displacement metric*, *absolute position metric*, *relative position metric*, see Figure 3, left), and the second type evaluates the orientation of an object (*rotation metric*, *orientation metric*, see Figure 3, right). These metrics evaluate the differences between the *effects* on the environment (objects) of the model and the imitator and do not consider the state or the actions of the agents. To evaluate both the movement and the orientation of an object, both metric types must be used. A weighted combination of more than one of the displacement metrics can be used, by a (convex) weighted sum of the displacement vectors that minimise each metric. For more details on these (and additional) effect metrics see [17].

The sub-goal granularity is given by finding the *critical points* in the trajectories of the manipulated objects. A critical point occurs when the direction of the captured trajectory and/or the orientation of an object changes by more than a certain threshold.

III. Experiments

In the experiments presented in this section, a human demonstrator arranges three block objects on the 2D workspace surface. The captured effects are shown in Figure 4, left. This single demonstrated task is used for all the following experiments which illustrate the capacity of the JABBERWOCKY system to generalize the matching of a demonstrated task across different starting object configurations.

Here, the metrics and granularity have been selected

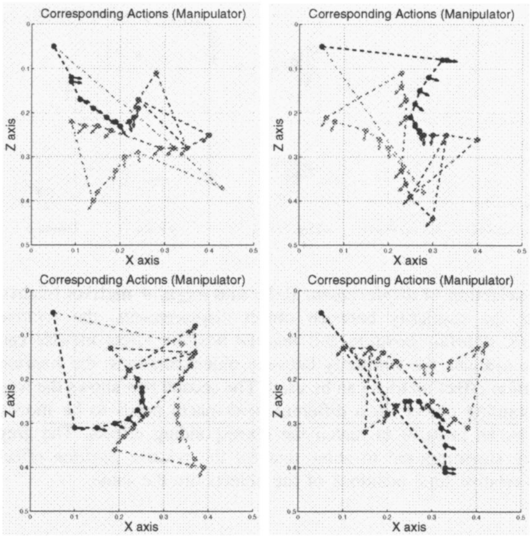

Fig. 6. **Corresponding action commands (using the *relative position* and *rotation* effect metrics) for dissimilar initial object positions.** Using the critical points shown in Figure 4 (right), starting from each of the initial configurations shown in Figure 5 (respectively), and minimizing the *relative position* and *rotation* effect metrics for all the objects, the imitating manipulator in each case must follow the continuous closed path (starting and ending at the left top corner of the workspace) shown as a dotted line. The line in drawn using a grey to black color gradient to indicate the direction of the path. Upon reaching an object, a small arrow indicates the orientation that the object must be rotated to.

in advance[1], and as can be shown (see [13], [16], [17] and below), depending on the choice, the character of the resulting matched effects can be very different. In the experiments, the *relative position* or *relative displacement* effect metrics are used to match the displacement effect aspects of the demonstration, and the *rotation* effect metric to match the angular effect aspects for all the objects in the demonstrated task. The sub-goal granularity is given by finding the critical points in the captured data (shown in Figure 4, right).

IV. Results

Given the above metrics and sub-goal granularity, and considering each of the several different initial object configurations (with dissimilar initial object positions) shown in Figure 5, the JABBERWOCKY system produces for each respective case corresponding action commands (as visualized in Figures 6) that when preformed by the imitator platform, result in 'successful' imitative behaviours (shown in Figures 7 and 8). In Figure 9, another example of initial object configurations (with dissimilar initial object orientations) and the resulting imitative behaviour is shown.

V. Discussion

A. Observations on Using the Relative Position Effect Metric

When the *relative position* effect metric is used, the relative position between the objects is conserved as they

[1] In the full system, the *what to imitate* module will provide a choice of metrics and granularity based on the task and context of the demonstration, although there might not always be a unique, "correct", choice.

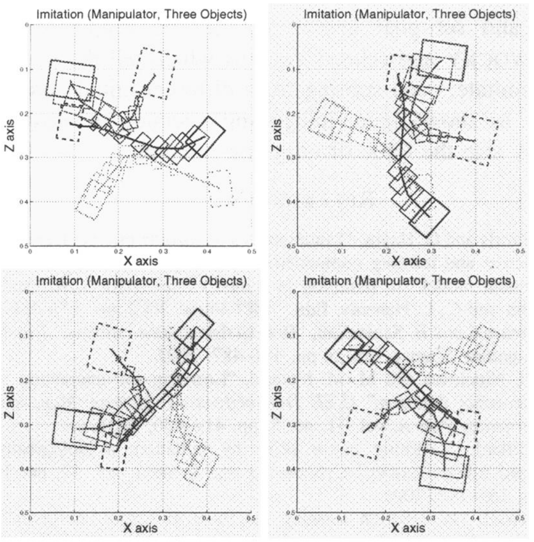

Fig. 7. **Resulting imitative behaviour using the *relative position* and *rotation* effect metrics.** Using, respectively, each of the corresponding action command sequences shown in Figure 6, the imitating manipulator shown in Figure 2 (right) performs the imitative behaviour (in Webots) and the captured results from the simulation are shown (representation as in Figure 4 and initial object configurations as in Figure 5).

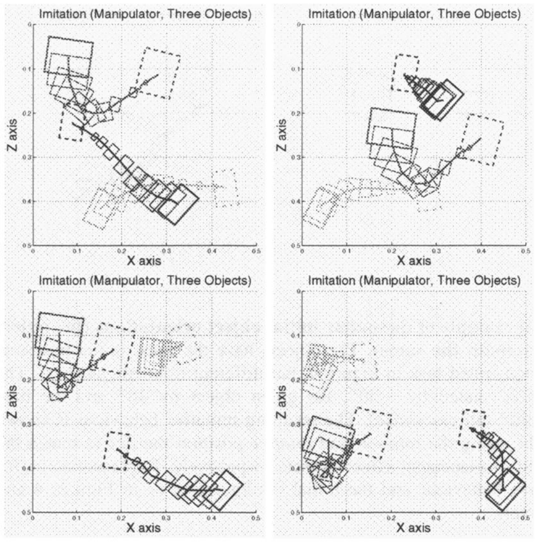

Fig. 8. **Resulting imitative behaviour using the *relative displacement* and *rotation* effect metrics.** Using the critical points shown in Figure 4 and starting, respectively, from each of the initial configurations shown in Figure 5. Representation as in Figure 4.

move around the workspace. This affects the resulting trajectories of the objects in each imitative behaviour, depending on the initial object configuration considered. For example, visually comparing the demonstration in Figure 4 (left) with the imitative behaviour in Figure 7 (lower right), one notices that the object trails are horizontally symmetric to each other, as a result of the initial object configuration used (rotated by 180° relative to the center of the workspace, Figure 5 lower right).

B. Observations on Using the Relative Displacement Effect Metric

When the *relative displacement* effect metric is used, the dissimilar initial object positions have a greater impact on the resulting imitative behaviours. Compared with using the *relative position* metric as in Figure 7 where the relative positions in the workspace are conserved, the objects in this case need to be moved in such a way as to conserve the overall shape of their trajectories, transposed according to the different initial positions. The relatively small size of this particular workspace, and the proximity of the objects to each other (in the initial positions given in Figure 5) are hindering the 'exact' matching of effects according to the metric. Nevertheless, a solution to the correspondence problem that satisfies the given metrics at the given sub-goal granularity and adapted to the context (including the state of the environment) can be found for most cases.

In three out of the four cases (except the one shown top left in Figure 8) the object manipulation, while attempting to match the demonstrated task encounter either another object that is blocking the way, or the edges of the workspace (action commands not shown). In the current implementation, when an effect (object displacement or rotation) that matches the demonstration according to the given metrics is invalid due to the above reasons, the system will modify (if possible) this suggested effect so as to achieve the matching displacement (or rotation) up to the point that it becomes invalid.

In Figure 8 (top right) the blue object is initially blocked by the red object and its trajectory is not matched entirely. The red object is also blocking the green object in Figure 8 (lower left), and is in turn restricted later in the imitative behaviour by the left edge of the workspace. All objects in Figure 8 (lower right) are obstructed by the edges of the workspace.

The objects in Figure 8 (upper left) are not obstructed by other objects or the workspace confines, but the imitation behaviour is qualitatively different to the one shown in Figure 7 (upper left). Both imitative behaviours are successful as they solve their respective correspondence problem (using different metrics).

C. Observations on Using the Rotation Effect Metric

The experiments described in the previous two subsections also used the *rotation* effect metric, but the initial object orientations were the same as the object orientation in the demonstrated task (see Figure 5). Using the *rotation* effect metric to match the orientation of the objects (and starting from the same initial object orientations) results in the objects achieving the same orientation as in the demonstration task during the imitative behaviour, although the object trajectories are 'different' in terms of absolute position (see Figures 7 and 8).

In contrast, Figure 9 (left) shows an example of dissimilar initial object orientations and the resulting imitative behaviour (right), resulting in visibly matched rotational changes (but *not* matched orientation) using the *rotation*

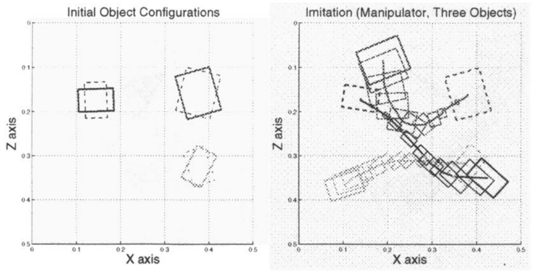

Fig. 9. **An example of dissimilar initial object orientations (the object positions remain the same).** The objects have the same initial positions as the demonstrated task in Figure 4, but different orientations (left). The red object is rotated by −30°, the green object by 45° and the blue object by 90° (all clockwise). The resulting imitative behaviour is shown on the right, using the *rotation* and *relative position* metrics to match the angular and displacement effect aspects respectively. Representations for the imitative behaviour and the initial configuration as in Figures 4 and 5, respectively.

metric. If matched orientation were required, the *orientation* effect metric should have been used instead.

The *relative position* metric was also used in the example in Figure 9, but since the initial object positions are the same in both the imitator's and demonstrator's workspaces (even though the object orientations differ), all the displacement effect metrics become equivalent.

VI. Conclusions

From the examples shown in the previous section it becomes apparent that the relative/absolute position and rotation of objects are important aspects of a demonstrated task to match (or not) according to effect metrics, depending on the state of the objects in the environment and the context.

Depending on the initial object configuration used, the JABBERWOCKY system is able to generalize based on the demonstrated task and produce appropriate corresponding action commands that when executed by the targeted imitator platform result in imitative behaviour, successful according to the sub-goal granularity and the metrics used to match the different aspects of the demonstration.

The examples presented in this paper illustrate that the JABBERWOCKY system is able to produce corresponding action commands for imitative behaviour that generalize with respect to dissimilar initial object configuration with different position and orientation of the manipulated objects (as compared to the demonstrated task). The action commands were here targeted for a single imitator platform (in simulation). The multi-platform targetability of the JABBERWOCKY system to map human demonstrated manipulations to matching robotics manipulations (in simulation) is shown in [17].

Robots programmed to learn human demonstrated tasks and skills by imitation will need mechanisms to match according to the different aspects demonstrated. Compared to a restrictive pre-programmed strategy, such a robot will be able to learn how to perform tasks in a more flexible way, adapting its execution according to the observed demonstrations by its human users. A wide selection of metrics and sub-goal granularity can be supported by JABBERWOCKY and related systems which tell the robots how to imitate, even starting from different initial configurations, as shown here. Integrating selection of *what* to imitate is also the topic of on-going research.

References

[1] J. Demiris and G. Hayes, "Imitation as a dual-route process featuring predictive and learning components: A biologically-plausible computational model," in *Imitation in Animals and Artifacts*, K. Dautenhahn and C. L. Nehaniv, Eds. MIT Press, 2002, pp. 327–361.

[2] C. Breazeal and B. Scassellati, "Robots that imitate humans," *Trends in Cognitive Science*, vol. 6, pp. 481–487, 2002.

[3] M. N. Nicolescu and M. M. Matarić, "Learning and interacting in human-robot domains," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 31, no. 5, pp. 419–430, 2001.

[4] A. Billard, "Learning motor skills by imitation: a biologically inspired robotic model," *Cybernetics and Systems*, vol. 32, no. 1-2, pp. 155–193, 2001.

[5] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in Cognitive Sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[6] Y. Kuniyoshi, M. Inaba, and H. Inoue, "Learning by watching: Extracting reusable task knowledge from visual observations of human performance," *IEEE Trans. Robot. Automat.*, vol. 10, pp. 799–822, November 1994.

[7] K. Dautenhahn and C. L. Nehaniv, "An agent-based perspective on imitation," in *Imitation in Animals and Artifacts*, K. Dautenhahn and C. L. Nehaniv, Eds. MIT Press, 2002, pp. 1–40.

[8] C. L. Nehaniv and K. Dautenhahn, "Mapping between dissimilar bodies: Affordances and the algebraic foundations of imitation," in *Proceedings European Workshop on Learning Robots 1998 (EWLR-7), Edinburgh, 20 July 1998*, J. Demiris and A. Birk, Eds., 1998, pp. 64–72.

[9] ——, "Of hummingbirds and helicopters: An algebraic framework for interdisciplinary studies of imitation and its applications," in *Interdisciplinary Approaches to Robot Learning*, J. Demiris and A. Birk, Eds. World Scientific Series in Robotics and Intelligent Systems, 2000, pp. 136–161.

[10] ——, "Like me? - measures of correspondence and imitation," *Cybernetics and Systems*, vol. 32, no. 1-2, pp. 11–51, 2001.

[11] ——, "The correspondence problem," in *Imitation in Animals and Artifacts*, K. Dautenhahn and C. L. Nehaniv, Eds. MIT Press, 2002, pp. 41–61.

[12] C. L. Nehaniv, "Nine billion correspondence problems and some methods for solving them," in *Proc. Second International Symposium on Imitation in Animals and Artifacts – Aberystwyth, Wales, 7-11 April 2003.* Society for the Study of Artificial Intelligence and Simulation of Behaviour, 2003, pp. 93–95.

[13] A. Alissandrakis, C. L. Nehaniv, and K. Dautenhahn, "Imitation with ALICE: Learning to imitate corresponding actions across dissimilar embodiments," *IEEE Trans. Systems, Man & Cybernetics: Part A*, vol. 32, no. 4, pp. 482–496, 2002.

[14] A. Billard, Y. Epars, S. Calinon, G. Cheng, and S. Schaal, "Discovering optimal imitation strategies," *Robotics and Autonomous Systems*, vol. 47:2-3, 2004.

[15] S. Calinon and A. Billard, "Stochastic gesture production and recognition model for a humanoid robot," in *IEEE/RSJ Intl Conference on Intelligent Robots and Systems (IROS)*, 2004.

[16] A. Alissandrakis, C. L. Nehaniv, and K. Dautenhahn, "Towards robot cultures? – Learning to imitate in a robotic arm test-bed with dissimilar embodied agents," *Interaction Studies: Social Behaviour and Communication in Biological and Artificial Systems*, vol. 5, no. 1, pp. 3–44, 2004.

[17] A. Alissandrakis, C. L. Nehaniv, K. Dautenhahn, and J. Saunders, "Achieving corresponding effects on multiple robotic platforms: Imitating using different effect metrics," in *Proc. Third International Symposium on Imitation in Animals and Artifacts – Hatfield, UK, 12-14 April 2005.* Society for the Study of Artificial Intelligence and Simulation of Behaviour, 2005.