Comprehension of Operating a Robot by Enactive Learning: Exemplary Approaches with Programming-by-Demonstration

Volker Schmirgel* Uwe E. Zimmermann* Eldad Yechiam† Thomas Hulin‡ Carsten Preusche‡
(*) KUKA Roboter GmbH, Germany
(†) Faculty of Industrial Engineering and Management, Technion, Israel
(‡) Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Germany
E-mail: VolkerSchmirgel@kuka-roboter.de, UweZimmermann@kuka-roboter.de,
yeldad@techunix.technion.ac.il, Thomas.Hulin@dlr.de, Carsten.Preusche@dlr.de

Abstract

Robot Programming-by-Demonstration demands two skills from the user, understanding the task that is to be programmed, and comprehending how to use the robotic system. This article is about training the user those skills that are necessary for the latter requirement. The robotic system is based on the DLR/KUKA light-weight robot, which allows direct human-machine interaction and compliant motions. An augmented reality and a virtual reality setup are presented that aim to demonstrate and evaluate skills transfer of two different sub-tasks of this system: avoiding robot singularities and setting correct compliance parameters. For this purpose training accelerators are introduced for (1) visualising robot singularities, (2) exploring robot singularities, and (3) feeling compliance parameters. An evaluation procedure for these accelerators is suggested.

1 Introduction

The state of the art in programming industrial robots is cumbersome writing of many lines of code usually carried out by experts specially trained in robot programming [3]. A new programming paradigm intends to enable robot programming in an interactive way by "taking the robot by the hand" and showing it what to do. This paradigm is called Programming-by-Demonstration (PbD) [4].

The idea behind this paradigm is that a worker who knows how to perform a task can program a robot simply by executing the same task, but with the robot in his hands. By guiding the robot with his own hands the worker is intuitively "demonstrating" how to solve the task. Ideally, not a single line of code needs to be programmed. The considered novice users of PbD should "demonstrate" those tasks that they are experts in. Since they do not have skills on robot programming, the key issue of this article is: How can users learn using PbD?

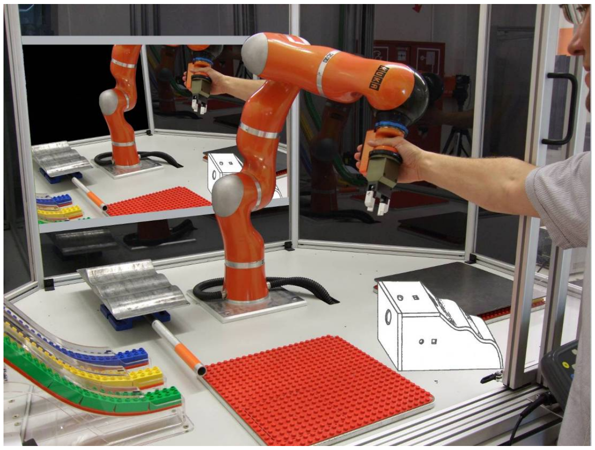


Figure 1: Programming-by-Demonstration system based on the DLR/KUKA light-weight robot.

With the new DLR/KUKA light-weight robot technology, being equipped with numerous internal sensors, the PbD paradigm can be introduced from research into commercial standard robotic applications. Due to this sensory equipment, including torque sensors in each joint, this robot is capable of being operated in compliant motion. Since the worker interacts with the robot in a direct physical way (see Fig. 1), new skills are needed to cope with the technical system [17]. Furthermore, some of the skills necessary for operating robots nowadays are also important for programming a robot with

PbD. Due to the fact that the robot in use is capable of interacting with the operator and that it can provide haptic feedback, the robotic system itself is used as a training system.

PbD can be used in different applications and with various process tasks. This shows that PbD itself is very complex and has high variation. As it is not possible to cover all aspects and all skills involved in the presented skill transfer system, only two sub-tasks have been selected to be evaluated on skills transfer scenarios:

- avoiding robot singularities, and
- setting compliance parameters.

Singularity avoidance means being able to avoid robot positions from where it cannot be moved further in certain directions [6]. This sub-task can be analyzed analytically allowing evaluation of user performance. The other scenario that has been selected is setting appropriate compliance parameters for the robot, such that the robot is able to perform a certain task. This scenario is much more sophisticated since performance parameters are unknown. Furthermore, this makes evaluation more challenging.

The following sections give an overview of a skill training demonstrator for comprehending the operation of a new robotic system. The demonstrator takes into account the SKILLS Unified Framework approach, with specific emphasis on multimodal feedback and enactive learning. Section 2 gives an overview of some tools for skills transfer and their possible drawbacks. Section 3 describes two setups used for demonstrating skills transfer, as well as training tasks that are foreseen to be performed on them. Three possible accelerators for training and skill transfer are introduced in section 4, and a plan for evaluating them is presented in section 5. Section 6 summarizes the main issues of this work.

2 State-of-the-Art in Visual and Haptic Guidance

The most intuitive way in which feedback in a PbD scenario can be given is by visual guidance. For example, arrows can be used to point out positions leading out of singularities. Visual graphs can be used to show the user the current compliance level and the deviation between current and requested levels. In PbD visual feedback naturally supplements the existing haptic feedback when the user guides the robot into place. The current section briefly reviews the literature on visual and haptic guidance.

Two main theories postulate that using the visual and auditory channels for presenting information improves skill acquisition compared to the use of a single modality [11, 12, 19]. Both theories are based on Baddeley's model of working memory [1]. According to this model, working memory is limited in capacity, but includes two processors, auditory (verbal) and visual, which can be used simultaneously. The first theory is the Cognitive Theory of Multimedia Learning [11] which argues that having two modalities available during training enables a trainee to build two different mental representations - a verbal model and a visual model - which enriches the stored memory. This theory is supported by findings in multimedia studies showing that adding pictures and animations to vocal narrations improves success rate in subsequent knowledge tests (e.g., [12]).

A second theory supporting the use of multiple modalities during training is Cognitive Load Theory [19, 20], which asserts that using multiple sensory channels reduces the cognitive load on working memory. These arguments were also supported by research findings [5, 14]. Together, the two theories support the use of multimodal training including visual guidance, along with information received by other modalities.

Questions about the use of visual guidance arise from the old literature on experimental psychology. One of the most famous experiments that has demonstrated the disadvantage of passive visual guidance in spatial tasks was a study by Held [8], which involved two kittens that were put in a device where one controlled it's visual environment by moving towards or away from objects, whereas the other cat's visual environment was entirely controlled by the first cat's movements. The results showed that the cat that was able to actively explore the environment had much better performance in spatial tasks. While Held's study focused on the global importance of active exploration during the development process, it is likely that a similar effect appears in the micro environment of a skill acquisition process where a trainee is passively (visually) guided by the trainer. Indeed, this was confirmed in a series of recent studies conducted in the SKILLS project.

For example, Gavish, Yechiam and Kallai [22] conducted an experimental study in which trainers instructed trainees on how to perform a 3-D puzzle. This was done in two conditions: Vocal guidance, where only vocal instructions were possible, and vocal guidance with mouse pointing (where the trainer could also use a mouse to point out positions on the trainee's screen). The results showed that $90\,\%$ of the trainers used the mouse pointing option when it was available. However, the use of the mouse pointer, while reducing mental load during training, drastically impaired trainees' success rate and performance speed. Specifically, it led to a performance decrement of about $50\,\%$ in the proportion of successful completers and in response time. These results suggested that the use of multimodal training using visual guidance should be re-evaluated.

Still, there are conditions where visual guidance is necessary because visual feedback cannot be easily supplemented by alternative modalities. Singularity avoidance in PbD is a case in point. It is very difficult to provide information in six degrees of freedom in a non-visual manner. This led the current design process of PbD to consider avenues of reducing the possible harmful effects of visual guidance. The negative effect of visual guidance is posited to stem from passively following the trainee's request rather than actively exploring the environment [22]. Therefore, a training protocol necessitating the trainee to actively explore the task environment after getting visual feedback is predicted to offset this negative effect [21]. The next sections provide the details of the selected arrangement of visual and haptic cues.

3 System

PbD in this article is based on the KUKA/DLR light-weight robot (LWR). The LWR is a revolute joint robot, with integrated electronics comprising torque and position sensors [9]. This sensory equipment enables compliant behaviour and opens the door to the sensitive area of direct human interaction. The LWR has a load-to-weight ratio of $1:1$ such that it can handle objects of a weight up to $14\,kg$. With its seven serially linked joints, this robot possesses redundant kinematics that allow null-space movement, which is valuable for avoiding collisions and optimizing the robot's configuration [18].

Two basic technology platforms are used: One is using the real robot performing a real task (On-Site Setup). The user is provided with augmented reality (AR) aids. The other is using the robotic arm in a virtual reality (VR) haptic and visual environment to perform simulated tasks (Training-Center Setup). Thus, with the choice of using two training platforms, the advantages of both are combined, i.e. with a real robot (On-Site Setup) the PbD task is realistic, and with virtual reality technology maximum flexibility is achieved, in terms of modifying the scenario, and adapting visual or haptic guides. The following two subsections describe each platform separately:

3.1 On-Site Setup

The On-Site Setup of the PbD-Demonstrator is an AR scenario where virtual (visual and haptic) information is augmented to a real life setup (see Fig. 2). The AR scenario is set up in the actual working cell with a DLR/KUKA light-weight robot and task related equipment. It consequently features all the details of a real working setup. With this setup we examine how additional sensory information can be used to enhance learning in real life environment.

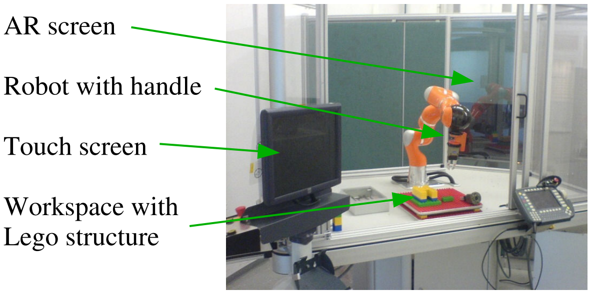


Figure 2: The On-Site Setup with touch screen for interaction and instructions, robot with handle, workspace with Lego structure and a big screen for AR.

The task is to place a Lego brick, which is already in the gripper of the LWR, into a notch of a given Lego structure (see Fig. 3). The brick is placed in front of the structure close to the desired final position. The following motion is crucial, as it inserts the brick into the notch and then moves it inside the notch to a position where the brick is exactly above the desired position, and where the brick must be pressed down.

For compliant motion a trajectory and compliance parameters are necessary. The trajectory for this motion is already provided by the system. The user is supposed to set the compliance parameters of this motion appropriately, which means the robot successfully reaches the desired position without getting stuck on the way or damaging the structure.

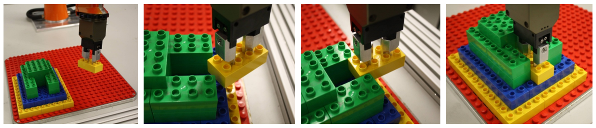


Figure 3: On-Site Setup task: Set parameters for fitting Lego brick in a notch.

The focus of research with this setup concentrates on the benefit of a haptic accelerator which is described in detail in section 4.3. Beyond that, the setup provides further sensory information and assistance to the user. Subsequently, some exemplary features are sketched. For simplifying the task, the system can limit the set of changeable parameters to those, which are necessary to solve the task successfully. The user may get visual guidance/assistance on the AR screen, e.g. a virtual coordinate system. While executing the process, the user may watch occurring forces and torques of the robot

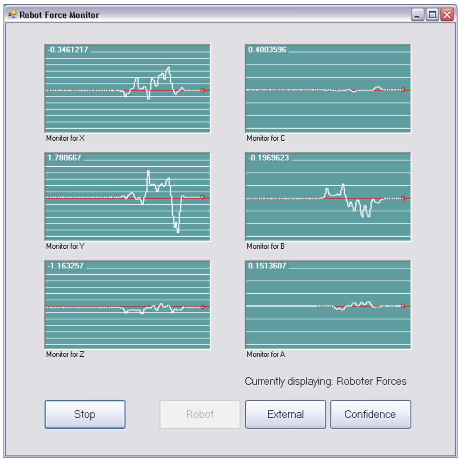


Figure 4: While executing the process, the user may observe occurring forces.

(see Fig. 4). On the plots optimal forces (from expert's solution) may be displayed. After performing the training, a score that is calculated of several measurements (e.g. number of steps, required time, deviation to ideal solution ...) may be displayed to the user. The trajectory for executing the process may be selected to vary the peg-in-hole task: it is possible to choose it from several predefined trajectories (expert's trajectories) as well as to teach-in a self defined trajectory.

3.2 Training-Center Setup

The Training-Center Setup of the PbD-Demonstrator is based on VR technology to simulate different cases taken from real life and can be easily augmented with real components of the use case. The core component is the DLR/KUKA light-weight robot being used as haptic device, but the methodology will also allow for using other interfaces within this setup. Other technical components of this setup comprise the visualisation engine InstantReality from Fraunhofer IGD [2], and the haptic algorithm VPS [16] for computing collision forces and torques in the virtual scene. To realistically display these forces to the user, the haptic control has been designed following the stability condition for haptic rendering [10, 15, 7].

Time-delay is critical for skills transfer, if it differs in the virtual world from that in the real world. In this setup time delay for 3d visual feedback is within one sampling period for an update rate of 60 Hz, so that this additional delay can be neglected. The round-trip delay in the haptic loop has been determined to be 6 ms.

This delay is limiting the stiffness of the contacts in the virtual world. Thus, also the coupling between the real and the virtual robot is limited at approximately $5000\,N/m$ (a higher stiffness would result in an unstable system behaviour [10]). This limitation causes a lag of the virtual robot's position with respect to the real one (see Fig. 5). During free-space movements, the resulting position difference is less than $1\,mm$, or in terms of delay less than $3\,ms$ (in the region where the robot is moving, i.e. $t < 0.055\,s$). Given the duration of these delays, it is not expected that delayed feedback at this level will have negative effects on training acquisition (see also [13]).

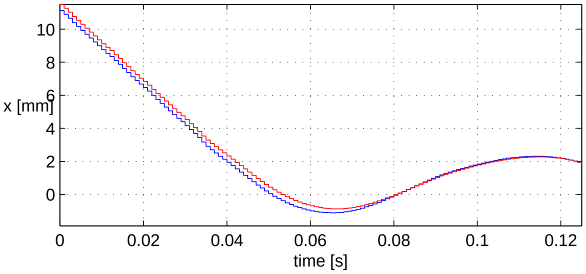


Figure 5: Position of the virtual robot (red line) is lagging behind those of the real robot (blue line) during training task.

The purpose of this setup is the transfer of skills related to the handling of the robot (robot skills) to a novice user. The use cases can be adopted to concentrate on specific aspects of certain skills needed. For the selected sub-task "singularity avoidance" a Lego game environment has been developed in the VR (see Fig. 6). The goal of the training task is to train the user to understand the effects of robot singularities and to avoid them.

In the Lego game the user should build a predefined Lego structure with a virtual LWR, while trying to avoid singular robot configurations. The virtual robot's end-effector is coupled to the real robot being used as haptic device, such that the user can move and rotate the virtual robot, and feel forces and torques that occur in the virtual world. Thus, not only collisions between the virtual robot and the virtual environment are displayed, but also the dynamics of the virtual robot itself. This is essential to feel the effect of robot singularities.

In the VR a red blinking Lego brick is indicating the pick- and place-positions. The trajectories between these positions can be freely chosen by the user. During the game different means of help can be activated. Some of them are selected as accelerators and described in the following section.

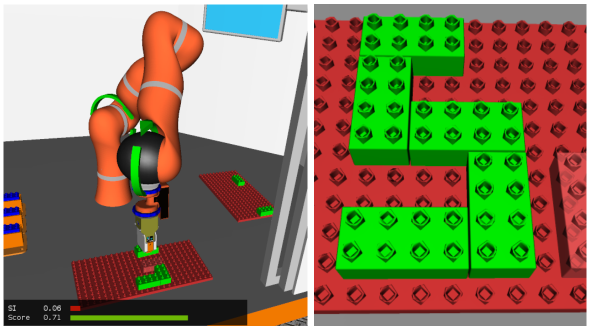


Figure 6: In the Training-Center Setup the trainee must use the virtual robot to build a predefined Lego structure.

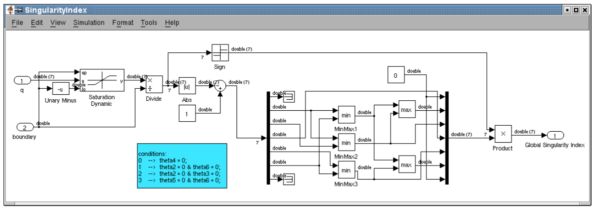
4 Accelerators

The term accelerator refers to a methodology that increases the speed of skill acquisition. In the training of Programming-by-Demonstration three potential accelerators were chosen based on the following general criteria: 1) involve sub-tasks that are difficult for many novice programmers to acquire 2) sub-tasks that are general and appear in most tasks of robotic programming 3) sub-tasks that are unique to the domain of robot handling. Two accelerators focus on the sub-task of singularity avoidance and involve a visual guidance technique and a training protocol for encouraging participants to actively explore their environment (sections 4.1 and 4.2). The final third accelerator focuses on the task of setting compliance parameters (section 4.3).

4.1 Singularity Index and Performance Indicator

The singularity index is a value that describes the proximity of a robot to its closest singular configuration. Different methods can be used to calculate such an index. The implemented method is described in the following lines. Its main advantages are (1) that it gives a singularity index for each robotic joint instead of one global index, (2) the singularity index indicates the direction for each joint to avoid the singularity, and (3) it can be easily extended in order to recognize sub-singularities, which are singularities of a part of the robot (e.g. if the first three joints of the robot lose one degree-of-freedom).

The module for computing the singularity index is implemented as Matlab-Simulink block (see Fig. 7). It is connected to the visualization system, which uses rotating arrows (see Fig. 8) to indicate the direction the joint has to move in order to increase the distance to the singularity.



conditions:
0 --> theta4 = 0;
1 --> theta2 = 0 & theta6 = 0;
2 --> theta2 = 0 & theta3 = 0;
3 --> theta5 = 0 & theta6 = 0;

Figure 7: Simulink subsystem for calculating the singularity index of the LWR.

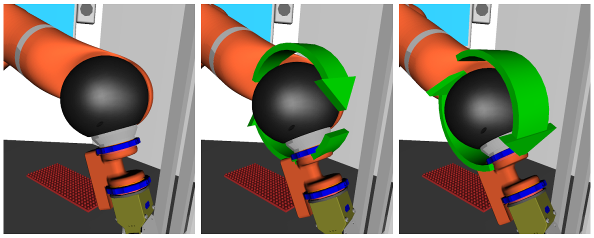


Figure 8: Green rotating arrows indicate the singularity index of each joint. The wider the arrow, the closer is the robot to a singularity.

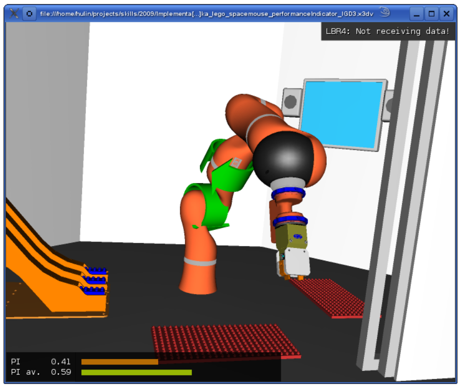
The singularity index is used by a performance indicator to indicate the trainee's performance during a task. Furthermore, it is a variable that is stored in the Training-Center Setup.

In order to communicate his success to the trainee during task execution, a performance indicator has been developed. Due to the fact that haptic feedback is already included inherently in PbD, using visual or audio "feedback indicators" suggests itself. Thus, the implemented performance indicator is realized as a graphical module in the visualization window.

In the current version it is possible to visualize two scores, numerically and graphically (see figures 9 and 10). Different parameters can be mapped to these two scores. For example, for the training of singularity avoidance, the first score of this performance indicator represents the current singularity index of the robot, and the second one a total score of the trainee.

4.2 Exploration of Singularities

It has been determined in a first step that there are four possible conditions for a singularity of the LWR, as shown in Fig. 11. In such a condition, one or two joints must be at zero position, while the other joints can be arbitrary. In the Training-Center Setup defining a starting configuration for the virtual robot is possible. At



LBR4: Not receiving data!

```
PI      0.41
PI av.  0.59
```

Figure 9: Visualization window of the Training-Center Setup, showing the performance indicator in the bottom-left corner, and rotating green arrows that indicate the direction of movement out of singular robot configurations.

```
SI      0.18
Score   0.82
```

Figure 10: Performance indicator is showing the singularity index and the score of the user.

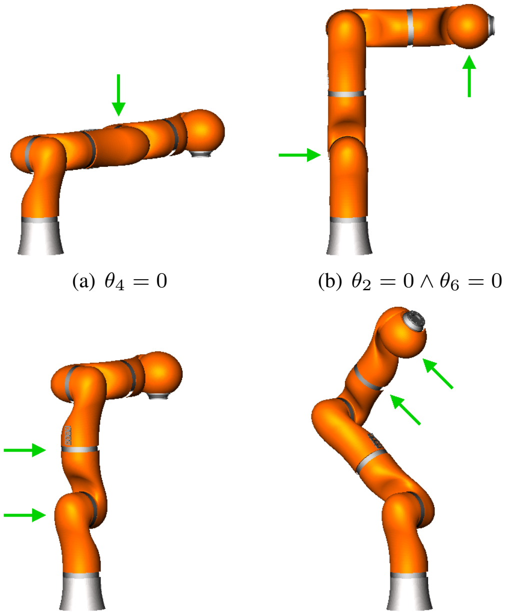
the same time it must be guaranteed that the real robot, which is used as haptic device, is in an initial condition where it has enough free space in all six Cartesian degrees of freedom.

In the starting configurations the virtual robot must be singular. The user may move the virtual robot via moving the haptic device. When starting to move the virtual robot from a singular position, one can easily explore the effect of singularities: the robot cannot move in certain directions anymore. The experiment's training protocol will involve having the users start in different singularities, get out of them, and re-set the system to these singularities. Thus, the user can explore all four robot singularities and their effect.

4.3 Haptic Feedback

This accelerator facilitates the setting of compliance parameters that is done before a robot makes a series of movements. It supports the acquisition of the cognitive model of compliance through a direct haptic feedback in the course of the robot movement.

There are six compliance parameters for Cartesian space, which define the translational and rotational stiffness. When starting the parameterization, the compliance parameters are set to an initial parameterization by the system. The initial parameters are not known to the user. He may activate the robot with the actual parameterization, grab the handle and "feel" the compliance of the robot (see Fig. 12). The user may then change the parameterization for each dimension with +/- buttons on the screen (see Fig. 13). The changes are immediately transferred to the robot's control system, so the user can "feel" the result of these changes by grasping the handle and exerting forces on the robot.



(a) $\theta_4 = 0$

(b) $\theta_2 = 0 \wedge \theta_6 = 0$

(c) $\theta_2 = 0 \wedge \theta_3 = 0$

(d) $\theta_5 = 0 \wedge \theta_6 = 0$

Figure 11: Singular configurations of the LWR. The green arrows indicate the robot joints that cause the respective singularity.

With this accelerator the user starts robot motion only when he thinks that the task can be accomplished with the compliant behaviour he has felt with his own hands. The user observes the robot executing the given

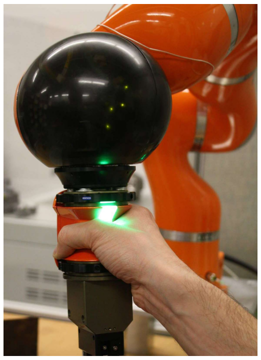


Figure 12: Haptic feedback. The user can "feel" the parameterized compliance of the robot.

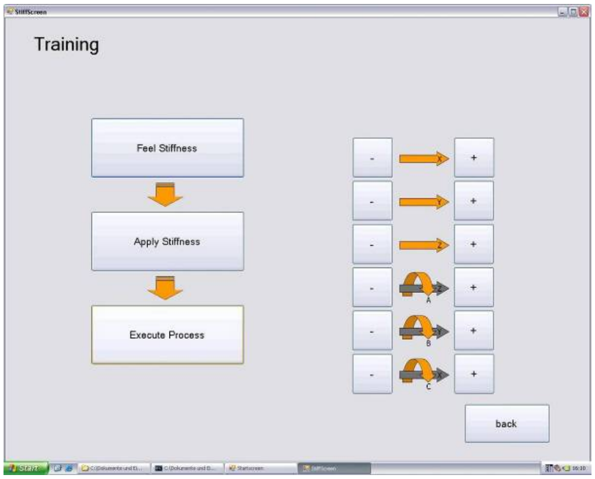


Figure 13: The user can adjust the six compliance parameters of the robot by pressing at the + and - signs.

task. He may subsequently alter the parameters and feel the result of his action again, in order to improve the performance of the robot.

5 Evaluation Plan

This section presents an evaluation plan that comprises the three potential accelerators of previous section. The accelerators focus on two sub-tasks which were found as critical based on the task analysis: singularity avoidance and setting of compliance parameters.

The challenge of singularity avoidance is to prevent the robot's posture from reaching configurations where it gets singular, i.e. it cannot be moved in certain directions anymore. To support the learning process, we use two accelerators: A visual pointer which directs the user to the required moves for avoiding the singularity, and a training protocol (called "Exploration of Singularities") which is aimed to offset the negative effect of "passive guidance" that appears to be a side effect of visual guidance. The complete training will start with the pointer as an easy first step, and then move to a stage where the participants explore the avoidance of singularities in various positions without using the pointer. Each of these two steps will be evaluated separately. The evaluation studies will have the training condition (with pointer / without pointer; with exploration training / without exploration training) as independent variable and performance (number of moves to complete a task, response time) as dependent variable.

The challenge of setting compliance parameters is to set them so that they will be optimal to the specific task at hand. For this reason, a firm grasp of the concept of compliance is required. The accelerator for this uses haptic feedback. In standard robotic application a user programs compliance levels and then observes the results. In the current system, the user will be provided with direct haptic feedback of the parameterized levels of compliance. The evaluation study will use the training condition (visual feedback; visual + haptic feedback) as the independent variable and performance (number of moves to complete a task, response time) as a dependent variable.

6 Conclusions

The current paper reviewed three potential accelerators for improving skill acquisition in the context of Programming-by-Demonstration. (1) A Singularity Index and Performance Indicator — a visual-based tool giving the user experiential feedback about the procedures needed to be undertaken when the robot configuration is close to a singularity. (2) A training protocol aimed to expose participants to singularities, which is referred to as "Exploration of Singularities". The general advantage of this second accelerator is in its non-visual nature. In previous SKILLS experiments visual guidance was shown to have some important side effect (e.g., [22]). (3) The final accelerator involves haptic feedback for setting compliance parameters. In this training protocol, users are encouraged to explore – haptically – the implications of setting various parameters. In this manner, users are directed not to rely on visual trajectories and other visual feedbacks denoting the value of different compliance parameters in a given task; but also on haptic sensations associated with different compliance levels. Because this additional haptic feedback makes use of active exploration, it is expected to facilitate the acquisition of the concept of compliance level (e.g., that certain compliance levels are appropriate for different tasks) and its implications.

7 Acknowledgments

We would like to acknowledge support for this work from the European Commission for funding of the Integrated Project IP-SKILLS-35005.

Furthermore, the authors of this article want to acknowledge the excellent cooperation of the people that have been involved in integrating the hard- and software for this demonstrator: Daniel Buchhard, Sascha Leyer, Andreas Summerer, Sabine Webel, Uli Bockholt, Katharina Hertkorn, Simon Schätzle, Holger Urbanek, Patrick van der Smagt, Philipp Kremer, Rainer Konieschke, Andreas Tobergte, Mikel Sagardia, Andreas Stemmer, Arturo Guzmán Carballido, Zhongyang Wu, Christian Liefhold, Nirit Gavish, Paul Smyth, Andrew Stoddart, Grigori Evreinov, and many more.

References

[1] A. D. Baddeley. *Human Memory: Theories and Practice.* Lawrence Erlbaum Associates: Hove, 1990.

[2] J. Behr, P. Dähne, Y. Jung, and S. Webel. Beyond the web browser - x3d and immersive vr. In *IEEE Virtual Reality 2007: Symposium on 3D User Interfaces (3DUI)*, USA, 2007. Fraunhofer IGD.

[3] G. Biggs and B. MacDonald. A survey of robot programming systems. In *Proceedings. Australasian Conference on Robotics and Automation*, 2003.

[4] A. Billard, S. Calinon, R. Dillmann, and S. Schaal. Robot programming by demonstration. In B. Siciliano and O. Khatib, editors, *Springer Handbook on Robotics*, chapter 59, pages 1371–1394. Springer, 2008.

[5] P. Chandler and J. Sweller. Cognitive load theory and the format of instruction. *Cognition and instruction*, 8:293–332, 1991.

[6] J. J. Craig. *Introduction to Robotics Mechanics and Control*. Addison-Wesley Publishing Company, Inc., USA, second edition, 1989.

[7] J. Gil, E. Sánchez, T. Hulin, C. Preusche, and G. Hirzinger. Stability boundary for haptic rendering: Influence of damping and delay. *Journal of Computing and Information Science in Engineering*, 6(1), March 2009.

[8] R. Held. Plasticity in sensory-motor systems. *Scientific American*, 213:84–94, 1965.

[9] G. Hirzinger, N. Sporer, A. Albu-Schäffer, M. Hähnle, R. Krenn, A. Pascucci, and M. Schedl. Dlr's torque-controlled light weight robot iii - are we reaching the technological limits now? In *IEEE Int. Conf. on Robotics and Automation*, pages 1710–1716, Washington D.C., USA, 2002.

[10] T. Hulin, C. Preusche, and G. Hirzinger. Stability boundary for haptic rendering: Influence of human operator. In *IROS2008*, pages 3483–3488, Nice, France, Sep. 2008.

[11] R. E. Mayer. Multimedia learning. *New York, NY: Cambridge University Press*, 2001.

[12] R. E. Mayer and R. B. Anderson. The instructive animation: Helping students build connections between words and pictures in multimedia learning. *Journal of Educational Psychology*, 84:444–452, 1992.

[13] S. Okamoto, M. Konyo, S. Saga, and S. Tadokoro. Detectability and perceptual consequences of delayed feedback in a vibrotactile texture display. *IEEE Transactions on Haptics*, 2(2):73–84, 2009.

[14] C. Penney. Modality effects and the structure of short-term verbal memory. *Memory and Cognition*, 17:398–422, 1989.

[15] C. Preusche, T. Hulin, and G. Hirzinger. *Human Haptic Perception: Basics and Applications*, chapter 33: Haptic rendering and control. Birkhäuser Basel, 1st edition, Oct. 2008.

[16] M. Sagardia, T. Hulin, C. Preusche, and G. Hirzinger. Improvements of the voxmap pointshell algorithm - fast generation of haptic data structures. In *53rd IWK - Internationales Wissenschaftliches Kolloquium*, Ilmenau, Germany, Sep. 2008.

[17] V. Schmirgel, U. E. Zimmermann, T. Hulin, and C. Preusche. *beyond movement*, chapter Position Paper: Human Skills for Programming-by-Demonstration of Robots, pages 144–167. Alinea editrice s.r.l., Nov. 2008.

[18] G. Schreiber, C. Ott, and G. Hirzinger. Interactive redundant robotics: Control of the inverted pendulum with nullspace motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 158–164, 2001.

[19] J. Sweller, J. J. G. van Merrienboer, and F. G. W. C. Paas. Cognitive architecture and instructional design. *Educational Psychology Review*, 10:251–296, 1998.

[20] J. J. G. van Merrienboer and J. Sweller. Cognitive load theory and complex learning: Recent development and future directions. *Educational Psychology Review*, 17:147–177, 2005.

[21] E. Yechiam, I. Erev, and D. Gopher. On the potential value and limitations of emphasis change and other exploration-enhancing training methods. *Journal of Experimental Psychology: Applied*, 7(4):277–285, 2001.

[22] N. Yuviler-Gavish, E. Yechiam, and A. Kallai. Multimodal training: The use of visual instructions in spatial tasks is a natural trap. *Manuscript submitted for publication*, 2009.