SEVENTH FRAMEWORK PROGRAMME

Project Acronym:	Xperience
Project Type:	IP
Project Title:	Robots Bootstrapped through Learning from Experience
Contract Number:	270273
Starting Date:	01-01-2011
Ending Date:	31-12-2015

XPERIENCE.ORG

Deliverable Number:	D5.2.2
Deliverable Title :	Scenario 1 demonstration (II): Robot demonstration on learning of generalized affordances
Type (Internal, Restricted, Public):	PU
Authors:	Giorgio Metta, Tamim Asfour, Ales Ude, Norbert Krüger, Giulio Sandini, Rüdiger Dillmann
Contributing Partners:	IIT, KIT, JSI, SDU

Contractual Date of Delivery to the EC:	31-01-2013
Actual Date of Delivery to the EC:	01-05-2013

Contents

Chapter 1

Executive Summary

This Deliverable describes the second year demos that have been developed in the context of WP5.2.

Chapter 2

Content of the Deliverable

2.1 Learning push affordance of objects from human demonstration using 3-D features

We have been investigating how object push affordances may be learned by performing human push experiments and recording experimental data using the Microsoft Kinect to gather 3-D object point cloud data, as well as the Polhemus Patriot electromagnetic tracking system to gather action trajectories. Here one of the central goals is to build on previous work on learning object affordances by developing richer representations to drive forward different types of learning with varying levels of supervision. To this end we have developed a 3-D feature descriptor that divides object point clouds into sub-parts and performs surface fitting in each of the sub-parts, from which shape features are extracted. This builds on previous work where 2-D object descriptors were used for affordance learning [4] and allows for a richer set of possible object push affordances to be learned and predicted. Rather than using pose-invariant visual features, as is commonly the case with object recognition, we ground the shape features of the object sub-parts with respect to their manipulation. That is, by using shape features that describe the surface of an object relative to the push contact point and direction. This use of a dynamic representation enhances the possibilities for learning about partially known or unknown objects, a goal which was proposed in the exploratory learning section of Workpackage 2.3. Our initial results were based on supervised learning using the 3-D feature descriptor to predict four labeled push affordance classes: left rotation, right rotation, forward translation and forward topple. This has resulted in a high degree of classification accuracy (96 %) when using five different unknown objects and five types of pushing actions over a small sample size of pushing trials (134). Our latest results include self-supervised formation of object affordance categories via feature clustering. Furthermore, we are also investigating training multiple models for accurate movement prediction and combining their predicted outcomes using a mixture of experts model. The results of this work are shown in the attached video **HumanPushExperiments.mov**.

Figure 2.1: Data acquisition for the extraction of push affordances. A magnetic tracker is attached to the finger.

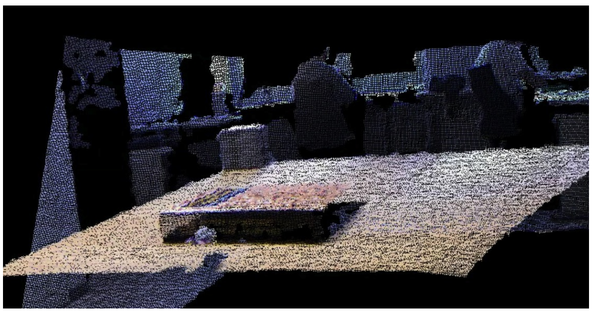

Figure 2.2: Kinect point clouds used for the extraction of action-dependent features.

2.2 Combining active visual learning of objects and reactive grasping based on haptic exploration

Learning the visual appearance and physical properties of unknown objects is an important capability for humanoid robots that interact with an open environment. The humanoid robot ARMAR-III is here shown to be able to discover, segment and grasp completely unknown objects in full autonomy, using its visual, manipulative and haptic capabilities. First, the robot generates hypotheses of possible objects based on the camera images of its active stereo vision system. To test these hypotheses and reliably segment them from the background, it then tries to move them. If thereby an object is discovered, its motion relative to the rest of the scene is exploited to segment it exactly and completely. Based on the now known object position and an estimation of its extent, we initiate an attempt to grasp it. The grasping itself is realized in a reactive manner using haptic feedback, allowing it to correct the grasp and adapt it to the actual shape of the object. Figure 2.3 gives a schematic overview of our approach. The results are shown in the attached videos **SegmentationAndGrasping.wmv** and **ObjectLearningWithForce.mov**.

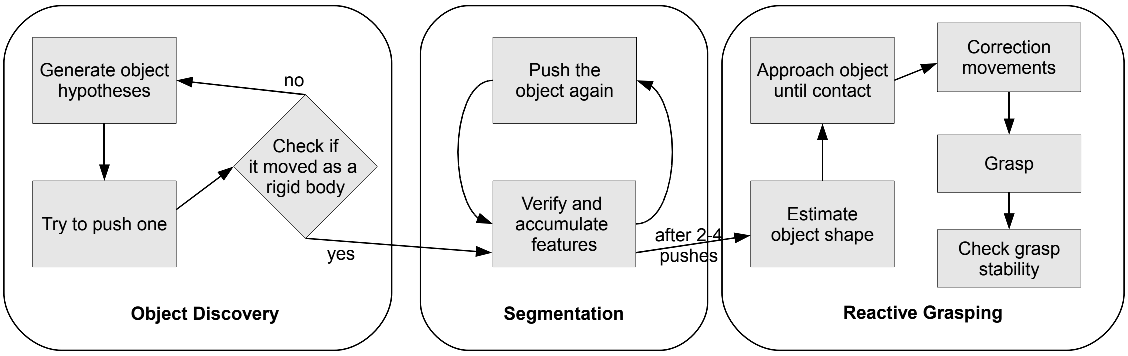

Figure 2.3: System overview showing the different stages of our approach for discovering, segmenting and grasping completely unknown objects

In the demonstrated experiments, the robot is positioned in front of a table on which several objects have been placed, as shown in Figure 2.4. The robot first looks at the table and creates object hypotheses based on the stereo images taken with the cameras in the head. Visual features such as Harris Interest

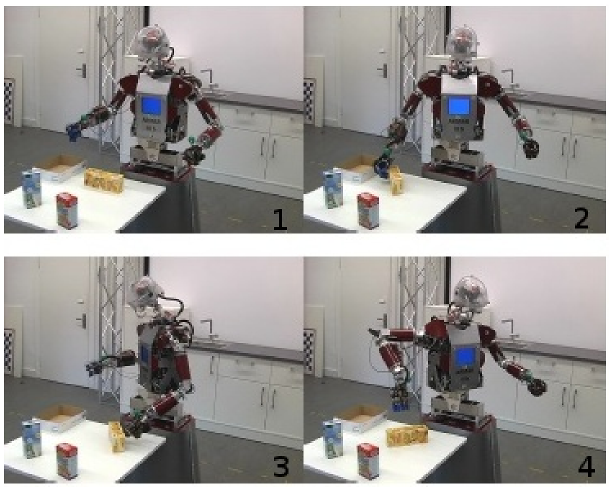

Figure 2.4: Frames from the video sequence. Object hypotheses are generated based on stereo images (picture 1). One of these hypotheses is then validated through several pushes (pictures 2-3). Based on the estimated object position and extent, a grasp attempt is initiated (picture 4).

Points and Maximally Stable Extremal Color Regions are extracted from the images; their 3D position is then determined through stereo matching.

Regular surface patches (planes, spheres and cylinders) are identified in the set of 3D points, and each of them can be considered a possible object. One of the potential objects is chosen for verification and pushed. When the hypothesis has been validated, i.e., the object has undergone a translation and rotation in 3D space, it is pushed again to accumulate all visible features belonging to it. During the later pushes, it is also moved towards a position that is well suited for the grasping attempt. Using the 3D positions of the features, the object position and the direction of its maximal extent are estimated.

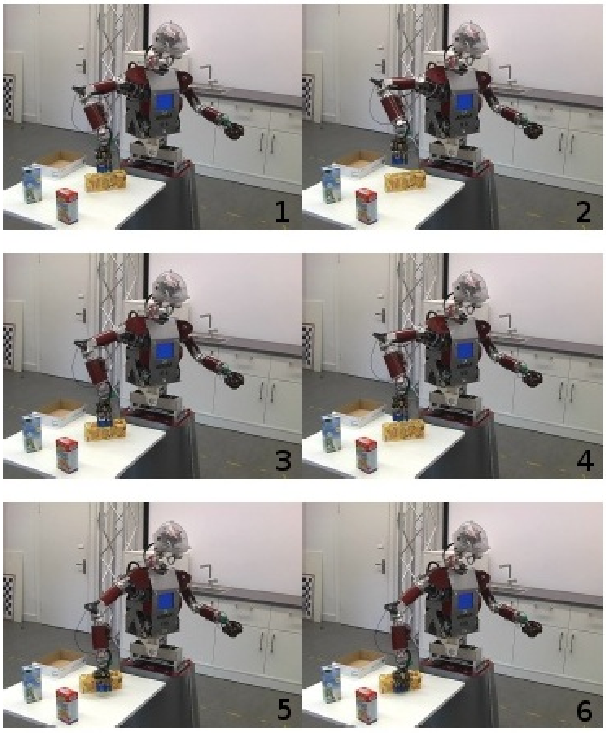

Figure 2.5: Frames from the video sequence. The robot reactively grasps the object using haptic feedback from different sensors in its hand and wrist.

The robot moves its right hand to a position above the verified points and turns the hand according to the orientation of the main object axis (figure 2.5). Then it starts to move the hand downwards towards the object, while continously checking for a contact with the sensors. If a contact is detected, the movement is stopped and a correction movement is executed, after which the robot starts moving the hand downwards again. This is repeated until a contact in the palm is detected. After that the hand is closed and the stability of the grasp is determined. If the grasp is stable the robot lifts the object and moves it to a position above a box, where it drops the object by opening the hand again. If the grasp is detected as not stable, the robot opens the hand again, and starts from the beginning.

2.3 Tool use generalization

Ongoing work regards learning of tool use and generalization of learned abilities across different tools that can be manipulated by iCub's dexterous hand. The scenario involves different target objects (i.e., octopus plush, toy car) and different simple tools, i.e., stick, hook, rake, hoe. The tools will be stored in a rack that allows the robot to grasp the handles firmly. The robot is taught explicitly the name of each tool and its position on the rack, as well as a set of actions (i.e, push the object away, pull the object towards oneself) that can be perfomed with all the tools, yielding to different results. The goal of the experiment is to have the robot autonomously determine which tool is more suited, and how it should be used, to accomplish a given task.

In a preliminary training phase, the robot is demonstrated how to perform the actions by means of kinesthetic teaching by an operator. Each sample action involves any of the target objects and any of the tools that are available. Each action is repeated many times, with the target object in different positions of the workspace. The 3D position of the target, estimated through stereo vision, is recorded together with the DMP parameters that describe the trajectory of the end-effector. Gaussian process regression is then applied to obtain a model that relates 3D position of targets (inputs) to DMP parameters (outputs) for each action [1].

Figure 2.6 shows the software architecture for this task, as implemented through a number of YARP modules. Vision, speech and motor-related capabilities are involved. The vision components detect objects in the scene and record their visual features, so that they can be recognized at all times during data collection; vision also provides an estimate of the 3D position of the object. Action recording and execution are motor abilities, while speech is needed to interact with human operators in a natural way. All these functionalities are coordinated by a set of "manager" modules, that mediate the timing and data flow for modules' execution. In the near future, self-coordination of modules will be enhanced, exploiting also the newly developed mechanism of "priority carriers".

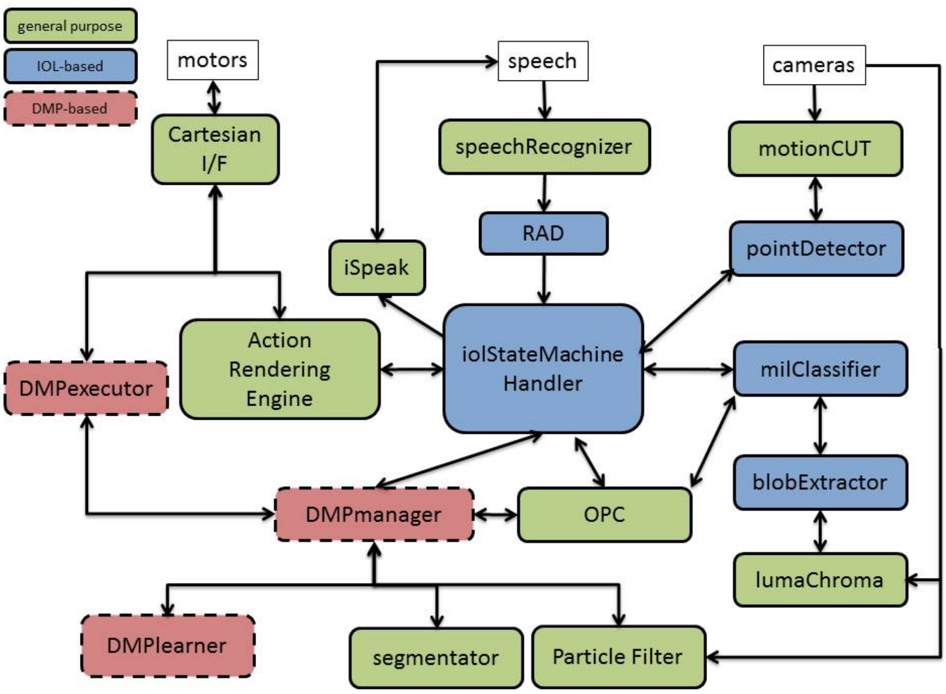

Figure 2.6: YARP architecture for the tool use generalization demo.

This first kinematics training step allows to gather the training data required to assess the affordance of each tool and target object. The robot performs all combinations of tool-action-object for different positions of the target object, while observing the state of the world and its own state. Wrist velocity and force, tool's approach direction (left/right/front/rear), object displacement (left-right, closer-away) are among the variables that are measured or estimated. A pictorial description of the scenario is given

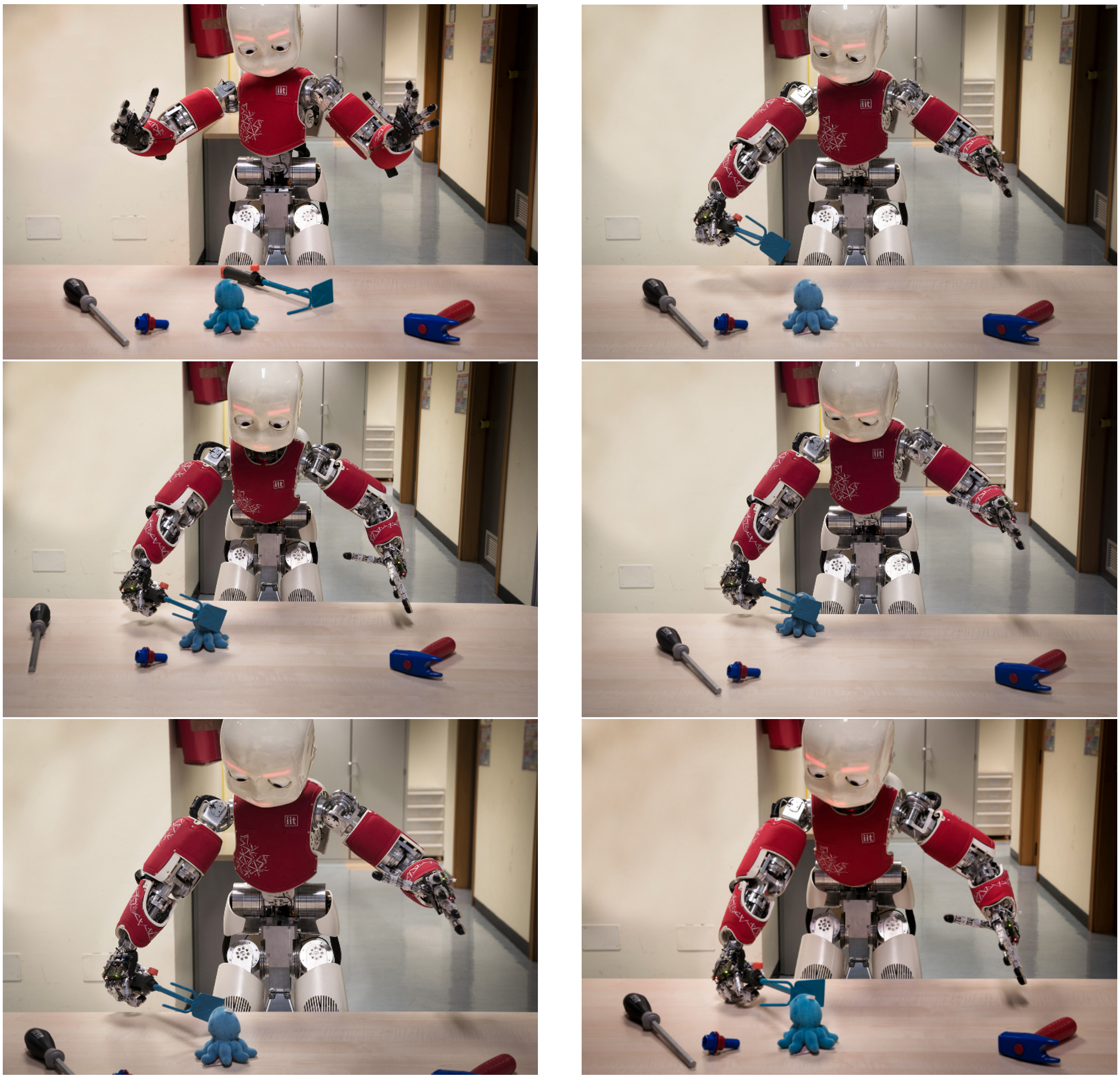

Figure 2.7: Scenario for the tool use experiment, and actions involved; Top row: left, initial situation; right, iCub grasps the toy hoe. Middle row: iCub pulls the octopus plush towards itself. Bottom row: iCub pushes the octopus away.

in figure 2.7, where the push and pull actions are perfomed by the robot using a toy hoe, and in figure 2.8, where the robot attempts the same actions using a stick, but with different results, as the pull action fails. Sensors' observations are used to derive explicit STRIPS rules that describe the domain of the tools affordances [3]. On such domain, structural bootstrapping can be applied to quickly learn the affordance of new objects and tools, and to plan which tools to use, and how, to reach a goal [2].

The kinematics training step is demonstrated in the attached video **icub-affordance-tool-learning.wmv**, along with parallel experiments on tool use in which actions' kinematics are refined through motor babbling, and the affordance of different tools with respect to the same action is determined through geometric reasoning.

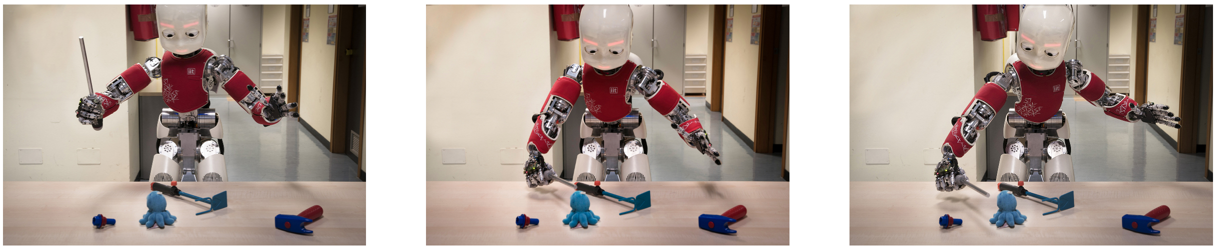

Figure 2.8: Push and pull actions perfomed using a stick. Left: iCub grasps the stick; middle, iCub pushes the object with the stick; right, iCub tries to pull the object with the stick, but it fails

References

[1] Denis Forte, Andrej Gams, Jun Morimoto, and Ale Ude. On-line motion synthesis and adaptation using a trajectory database. *Robotics and Autonomous Systems*, 60(10):1327 – 1339, 2012.

[2] Christopher Geib. Considering state in plan recognition with lexicalized grammars. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence – Cognitive Robotics*, 2012.

[3] Kira Mourão, Luke S. Zettlemoyer, Ronald P. A. Petrick, and Mark Steedman. Learning strips operators from noisy and incomplete observations. *Proceedings of the Twenty Eighth Conference on Uncertainty in Artificial Intelligence (UAI 2012)*, abs/1210.4889, 2012.

[4] Damir Omrčen, Aleš Ude, and Andrej Kos. Learning primitive actions through object exploration. In *2008 8th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 306–311, Daejeon, Korea, 2008.