An Affordance-Based Pilot Interface for High-Level Control of Humanoid Robots in Supervised Autonomy

Peter Kaiser[1], Dimitrios Kanoulas[2], Markus Grotz[1], Luca Muratore[2,3], Alessio Rocchi[2], Enrico Mingo Hoffman[2], Nikos G. Tsagarakis[2] and Tamim Asfour[1]

***Abstract*— In this work we present the concept of a pilot interface to control a humanoid robot on an abstract level in unknown environments . The environment is perceived with a stereo camera system and then simplified into a set of environmental primitives. Based on these primitives the interface proposes affordances to the pilot. Affordances are represented as certainty functions over the space of end-effector poses. The pilot operates the robot by selecting among proposed affordances and related action primitives, i.e. *Object-Action Complexes* (OACs). Before initiating execution, the pilot can review and revise the parameterization of the scheduled action primitive in a 3D reconstruction of the environment. The pilot interface proposed in this work has been implemented and evaluated on the humanoid robot WALK-MAN. With this work we also demonstrate the transferability of the perceptual concept, as our previous experiments have been performed using the humanoid robot ARMAR-III.**

I. INTRODUCTION

Despite impressive advances in the field, the cognitive capabilities of humanoid robots are still insufficient for autonomous employment of such systems in unstructured, human-centered environments. One of the key challenges involved is the autonomous perception of feasible ways of interaction between the robot and the environment. Actions of whole-body locomotion and manipulation are particularly important in this context as they lay the ground for a successful application of humanoid robots in real-world scenarios. Recent developments in humanoid robotics, especially regarding the *DARPA Robotics Challenge* (DRC), demonstrated that humanoid robots are already capable of reliably performing complex tasks when being conducted by a human pilot. These *supervised autonomous* interfaces leave difficult tasks to the pilot, such as high-level perception, action- and task planning, as well as general supervision.

In our previous work [1] and [2] we proposed a rule-based approach to identify possibilities of whole-body interaction in unknown environments based on the concept of *affordances* [3], targeting fully autonomous humanoid robots. This concept has been formalized in [4] as a hierarchy of affordance certainty functions. In this work we apply our ideas to the application of supervised autonomous robot control. We developed an affordance-based pilot interface, that potentially reduces the amount of teleoperation needed for the conduction of a humanoid robot in real-world applications. It also attempts to reduce the amount of prior, environmental knowledge necessary for the operation of a humanoid robot. The interface has been evaluated on the humanoid platform *WALK-MAN* [5] (Fig. 1), which successfully participated in the DRC finals in 2015. The underlying perceptual mechanisms have been developed and evaluated in several experiments using the humanoid robot *ARMAR-III* [6]. In the remainder of this section, we will briefly introduce the concept of supervised autonomy and discuss related work. Section II will introduce the fundamental concepts of the employed perceptual mechanisms . In Section III we will introduce the pilot interface and its components. Section IV provides an evaluation of the pilot interface based on a set of experiments carried out with WALK-MAN. Finally, Section V concludes the paper and discusses future work.

[1]Institute for Anthropomatics and Robotics, High Performance Humanoid Technologies Lab (H^2T), at the Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany. peter.kaiser@kit.edu

[2]Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego 30, 16163 Genova, Italy

[3]School of Electrical and Electronic Engineering, The University of Manchester, M13 9PL, UK

The research leading to these results has received funding from the European Union Seventh Framework Programme under grant agreement no 611832 (WALK-MAN).

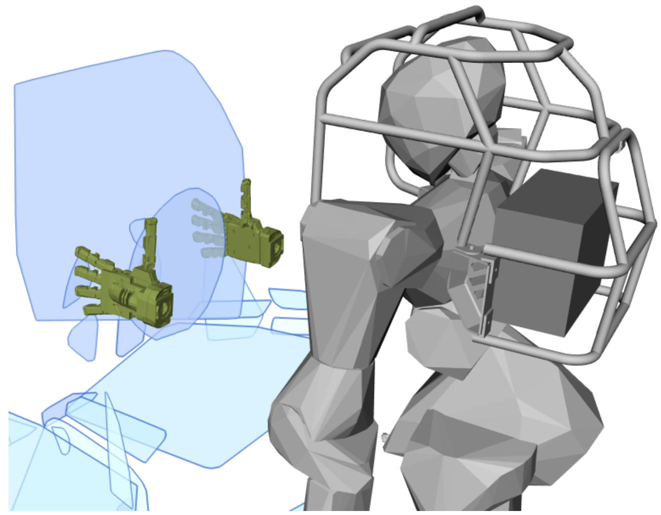


Fig. 1: The pilot interface visualizes the perceived scene in terms of environmental primitives and proposes end-effector poses for action execution, in this case for a valve turning task.

A. Supervised Autonomy

Humanoid robots can be roughly sorted into three categories based on the implemented degree of autonomy:

1) *Teleoperated robots* are remotely controlled by a human operator on a low level of abstraction, e.g. by set-

ting targets for joint angles or end-effector poses. Due to the complexity of humanoid robots, teleoperation is tedious for the operator and only partially applicable to complex operations.

2) *Autonomous robots* on the other end operate without human control or supervision, completely relying on the perceptive and decisive capabilities of the robot itself. Autonomy is largely regarded as the ultimate goal in humanoid robotics, and is still an active field of fundamental research.
3) *Supervised autonomy* is a compromise between the two extremes: The robot offers individual autonomous behaviors to a human pilot. The pilot selects among available behaviors and supervises the action execution. Other terms for such approaches include *shared autonomy* and *semi-autonomy*.

Although teleoperation tends to be tedious for complex robots, approaches exist for teleoperated control of humanoid robots. Some works approach this problem by moving away from joint-level teleoperation to either selective control strategies, e.g. [7], or by using inverse kinematics solutions, e.g. [8]. Other approaches exploit the structural similarities between humanoid robots and humans for mapping movements from a human operator to the robot using various motion capture technologies, e.g. [9], [10]. The robot software environment ArmarX [11], which is used throughout this work, offers basic joint-level and IK-based teleoperation functionality for a connected robot. Supervised autonomy was the predominant approach at the DRC. Examples for DRC-related pilot interfaces with supervised autonomous features include [12]–[17]. A comprehensive study of autonomy in the DRC can be found in [18]. Most of the DRC interfaces implement *object or affordance templates* that consist of parameterized models of environmental objects, e.g. a valve or a door, equipped with additional information regarding possible ways of interaction, e.g. rotation axes or grasp poses. The pilot interface proposed in [19] deduces applicable actions for known objects based on action templates defined in PDDL. A template-based approach was a feasible and reasonable choice for DRC since most environmental objects have been known beforehand. However, template-based approaches are not feasible in unknown environments, as there are no predefined models of environmental objects. In contrast to the above interfaces, we approach unknown environments.

B. The WALK-MAN Humanoid Robot

WALK-MAN [5] is an electrical motor driven 31 DoF humanoid robot with a total mass of $118\,\mathrm{kg}$ and a size of $1.91\,\mathrm{m}$. Its series elastic actuators can reach velocities and torques up to $19.5\,\mathrm{rad/sec}$ and $400\,\mathrm{Nm}$ correspondingly. Its head is equipped with a CMU Multisense-SL sensor that includes a stereo camera, a 2D rotating laser scanner, and an IMU. 6 DoF force/torque sensing is available at the end-effectors, while the hands are the underactuated Pisa/IIT SoftHands [20]. The robot software architecture is based on the YARP framework [21], while whole-body control and inverse kinematics are solved through the OpenSoT control framework [22]. Minimum jerk motions, as well as linear, composite, and circular trajectories are generated with trapezoidal velocity profiles using the KDL trajectory library [23].

The robot successfully participated at the DRC 2015, completing several outdoor tasks (e.g. driving and door opening) and most of the indoor tasks (e.g. valve turning and locomotion). Remarkable is that it was built in 11 months and completed only 4 months before the DRC finals. During the DRC, the WALK-MAN pilot performed manual object template planning through a Qt/ROS-based pilot interface [24]. The manipulation tasks were based on the raw point clouds acquired from the Multisense-SL, where the pilot was placing markers for environmental objects, e.g. the valve, door handles or end-effector target poses, in a 3D representation of the environment. In this work we propose a more abstract approach to a pilot interface based on the automatic extraction of affordances.

II. Whole-Body Affordances

The psychological concept of *affordances* [3] provides an intuitive way for describing the human perceptual process. In rough summary, it states that objects suggest possible ways of interaction to a perceiving agent based on object properties and the agent's capabilities. An overview of the theory of affordances and its application in different areas of robotics can be found in [25] and [26]. Although the techniques developed in this paper and in our previous work are not particularly limited to certain types of affordances, we concentrate on *whole-body affordances*, i.e. affordances that relate to whole-body actions. Such actions include large-scale manipulation, for instance pushing and lifting of large objects, or the establishment of contact with the environment for locomotion, for instance holding a handrail. One might argue if the affordances and actions demonstrated in this work fall into the whole-body category. However, we consider this work as a step *towards* a functional affordance-based pilot interface for whole-body actions.

A. Affordance Certainty Functions

In [4] we proposed a hierarchical formalization of affordances in terms of *affordance certainty functions*. An affordance a as understood in our approach corresponds to a certainty function Θ_a defined over the Cartesian product of the space $\mathcal{S}$ of end-effector poses[1] and the set Π of environmental primitives :

$$\Theta_a : \Pi \times \mathcal{S} \to [0, 1] \tag{1}$$

Affordance certainty functions Θ_* are multiplicative compositions of other affordance certainty functions and properties of environmental primitives transformed by sigmoid threshold functions. In this way, higher-level affordances are

[1] i.e. $\mathcal{S} = SE(3)$ for unimanual affordances and $\mathcal{S} = SE(3) \times SE(3)$ for bimanual affordances.

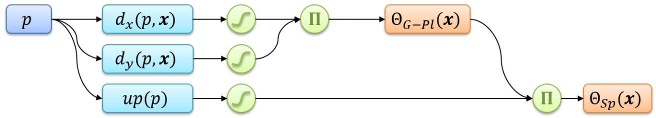
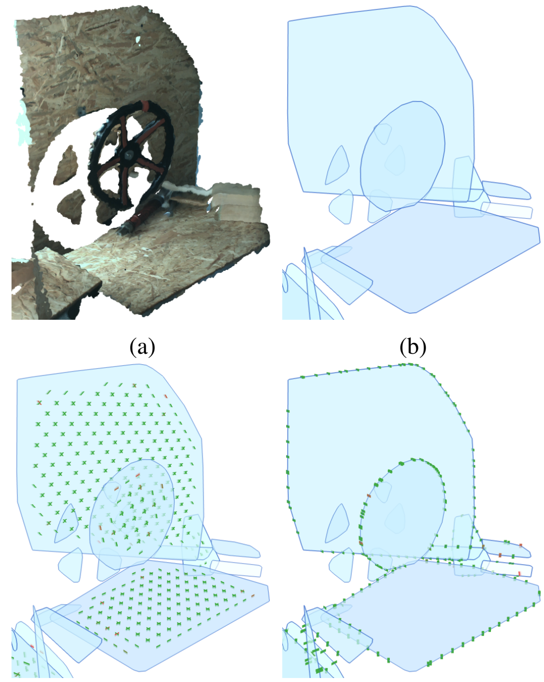


Fig. 2: The hierarchical composition of a unimanual *support* affordance $\Theta_{\text{Sp}}(\boldsymbol{x})$ for a primitive $p \in \Pi$ and an end-effector pose $\boldsymbol{x} \in SE(3)$. The support affordance is based on a *platform grasp* affordance $\Theta_{\text{G-Pl}}(\boldsymbol{x})$, combined with the additional primitive property $\text{up}(p)$ of horizontal orientation. The lower-level platform grasp certainty function $\Theta_{\text{G-Pl}}(\boldsymbol{x})$ essentially tells if the local neighborhood around the end-effector pose $\boldsymbol{x}$ on the primitive p, expressed using the distance functions $d_x(p, \boldsymbol{x})$ and $d_y(p, \boldsymbol{x})$, is large enough to fit the end-effector. Further details on the formalization of affordances in terms of affordance certainty functions are found in [4].

hierarchically composed from lower-level affordances, combined with additional conditions on the related primitives. Fig. 2 displays the composition of the affordance certainty function for the exemplary *support* affordance. Fig. 3 shows an exemplary point cloud of an industrial valve, together with the extracted environmental primitives and two visualized affordance certainty functions $\Theta_{\text{G-Pl}}$ and $\Theta_{\text{G-Pr}}$. The benefits of the hierarchical formalization of affordances in terms of affordance certainty functions include that the system inherently proposes suitable end-effector poses. These poses can serve as an initial parameterization of the subsequent execution of action primitives. Affordance certainty functions may include constant properties of the perceiving agent's embodiment. For instance, the perception of action possibilities may differ between ARMAR-III and WALK-MAN, since the their hand sizes differ. For further details on the process of affordance extraction, refer to [4].



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

Fig. 3: The perceptual pipeline applied to an exemplary stereo vision point cloud showing an industrial valve, a large back plane and a horizontal table beneath the valve. The individual figures show (a) the raw point cloud, (b) the extracted planar primitives, (c) a visualization of the certainty function $\Theta_{\text{G-Pl}}$ for platform grasps and (d) a visualization of the certainty function $\Theta_{\text{G-Pr}}$ for prismatic grasps. The marker color refers to the affordance certainty at the respective end-effector pose. Green means high certainty, red low certainty and certainties of zero have been omitted.

B. Object-Action Complexes

For action execution we follow the concept of *Object-Action Complexes* (OACs) [27]. OACs conceptually couple objects and actions into combined representations of sensorimotor behavior. They are defined as triplets

$$(E, T, M), \tag{2}$$

combining an execution specification E, a symbolic prediction function T and a statistical measure of success M. Affordances in this context can be regarded as preconditions for the existence of OACs, i.e. if an agent perceives an affordance a, a set of related OACs will be instantiated in the agent's memory. These OACs can subsequently be used for executing actions related to the perceived affordance a. Employing the concept of OACs has various benefits, including the persistent success measure M and the inherent support for higher-level planning components due to the enclosed symbolic prediction function T. As OACs couple objects and actions, they could also be used in the context of a template-based approach, as discussed in Section I-A. However, in the context of the proposed pilot interface, OACs represent a coupling between actions and environmental primitives and their availability depends on the existence of a related affordance. The concept of OACs is integrated in the robot software environment ArmarX via the *Spoac*[2] framework [28]. The execution specifications E of OACs are internally implemented as ArmarX statecharts [29] allowing a convenient implementation and modification of OACs via graphical tools. The system can also be integrated with symbolic planners, which will potentially allow even higher degrees of autonomy in the pilot interface.

As the perceptual system employed in this work is purely based on visual information and as the perceived environment is not known beforehand, the perception of affordances is prone to error. Perceived structures could for instance collapse once the robot starts to interact with them. To account for this fact, the result of the perceptual process in [4] is actually a set of *affordance hypotheses*, which are subject to validation by the robot or the pilot. The pilot interface itself does not differ between affordance hypotheses and affordances (i.e. validated affordance hypotheses). How-

[2]For *Symbolic Planning Integration Through Object-Action Complexes*

ever, after perceiving an affordance hypothesis a, the pilot is offered at least two related OACs: o_a for a direct action execution if the pilot is confident enough about the existence of a, or $o_{a,\text{val}}$ for a validation action that proofs or disproofs the existence of a.

III. The Pilot Interface

The affordance extraction pipeline discussed in the previous sections produces affordance certainty functions Θ_* from the perceived scene. These certainty functions can be evaluated in order to find possible and likely end-effector poses which can directly serve for OAC parameterization. The pilot interface proposed in this work is depicted in Fig. 4 and was designed with the following requirements in mind:

- The pilot should be given the choice between RGB-D, LIDAR and stereo vision data, if available, to account for individual shortcomings of these technologies.
- The pilot should be provided with visualizations of the underlying point cloud, the extracted primitives and the extracted affordance certainty functions.
- The pilot should be able to select among the affordance hypotheses that the system extracts from the scene.
- For each chosen affordance, the pilot should be able to select among the available related OACs.
- For each selected OAC, the pilot should see a visualization of an automatically proposed parameterization.
- The pilot should be able to adapt the parameterization if necessary.
- The pilot should be able to execute the selected OAC with the chosen parameterization.

In the remainder of this section we will discuss further details of the pilot interface and finally conclude by summarizing the pilot's workflow with respect to Fig. 4.

A. Action Parameter Selection

Each OAC requires an individual set of input parameters that the pilot has to specify. In the context of the proposed affordance-based pilot interface, the most elementary type of parameter is an end-effector pose. The affordance extraction process generates end-effector poses with assigned certainty ratings. However, further feasibility checks have to be carried out in order to provide the pilot with a restricted set of good suggestions for end-effector poses. One of the most elementary feasibility check is the reachability of the configured end-effector poses. Motion feasibility analysis on WALK-MAN is performed using the OpenSoT Previewer: the maximum cartesian error is tracked both during trajectory execution and for the final end-effector pose, and checked against a threshold for the maximum allowed task-space error. At the same time, collisions against the environment are checked using the *MoveIt!* library, which uses octomaps to perform robot-environment collision checking (see Fig. 5). The Previewer takes into account reachability, considering kinematics (i.e. joint limits), balancing constraints (i.e. CoM placement), workspace limits and environment constraints (i.e. obstacles).

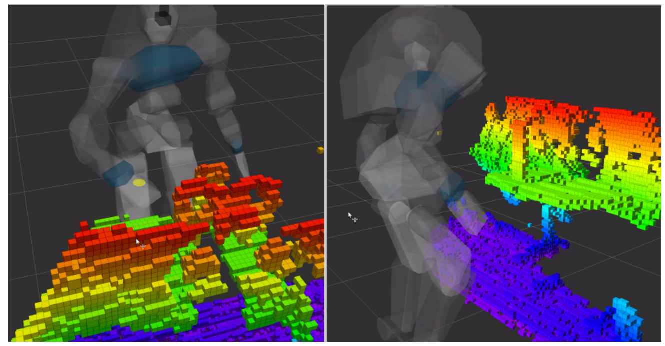


Fig. 5: An octomap is generated from the perceived stereo point cloud both for feasibility testing of end-effector poses as well as for collision checking between the environment and the WALK-MAN model. (rendered with *RViz*).

B. Workflow

In the following we will briefly discuss the workflow that the pilot follows when using the affordance-based pilot interface. The labels **A** - **G** refer to Fig. 4.

- **A** The pilot can control the affordance extraction pipeline. The pipeline can run continuously or in a one-shot mode, it can also be paused and resumed.
- **B** The pilot can configure fundamental parameters of the pipeline. This includes the point cloud source (RGB-D, LIDAR or Stereo Vision), the degree of cropping and the configuration setup of the segmentation and primitive extraction steps of the pipeline. The pilot chooses among predefined configuration setups.
- **C** The pilot can configure the visualization. Possible visualizations include the raw point cloud, the segmented point cloud, the extracted primitives and the extracted affordances.
- **D** The pilot sees a 3D reconstruction of the scene perceived by the robot, including a visualization of the robot itself. This scene can be freely rotated and zoomed.
- **E** After selecting a primitive by clicking, the pilot sees the affordances that have been automatically extracted by the system. Below each available affordance, the interface lists the related set of OACs that can be executed.
- **F** The pilot selects an OAC and sees a visualization of the OAC parameterization: Either a single end-effector pose or two end-effector poses in case of a bimanual OAC. The pilot can manually adjust the end-effector poses in the 3D environment if the proposed parameterization is not sufficient (see Fig. 6).
- **G** Once the pilot is satisfied with the OAC selection and its parameterization, it can be executed.

In the following section we will first discuss the evaluation scenario and the integration work needed to control WALK-MAN with the ArmarX-based pilot interface. Then

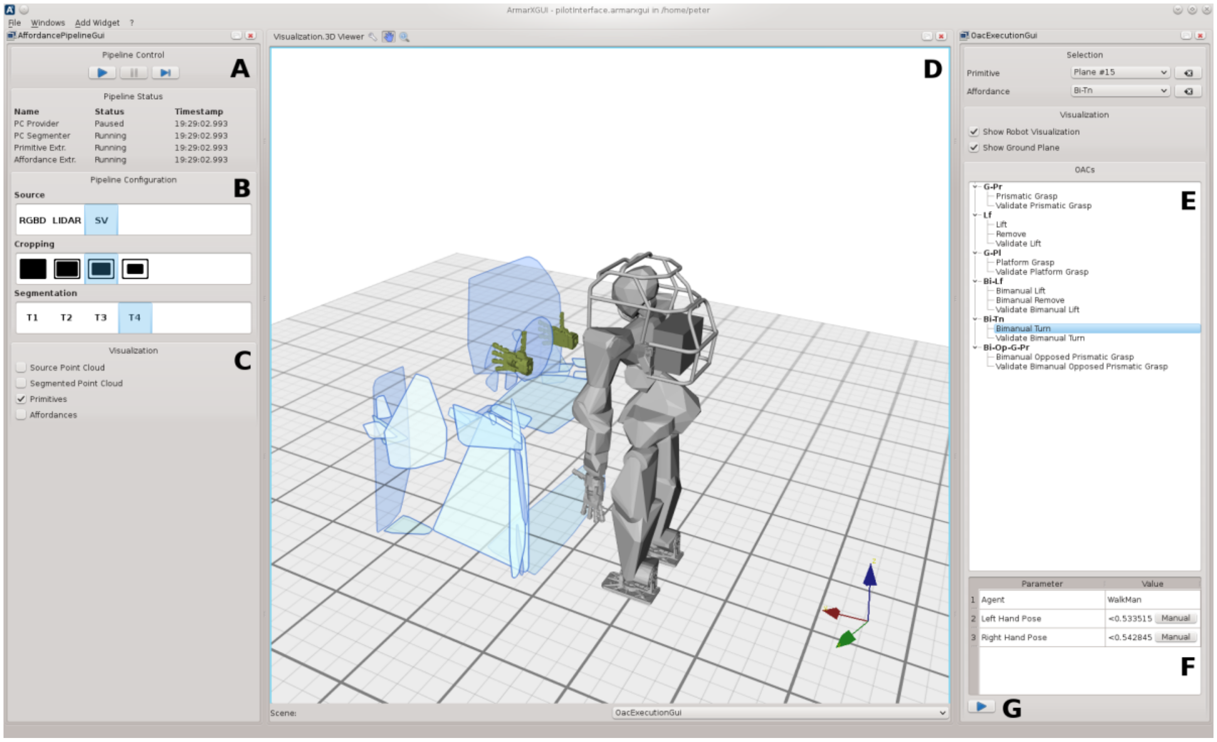


Fig. 4: The proposed pilot interface for high-level control of humanoid robots. See Section III-B for a detailed description of the individual components **A** - **G**.

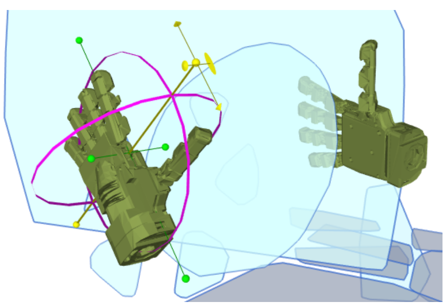


Fig. 6: The pilot is able to conveniently adjust the end-effector poses proposed by the affordance extraction system if necessary.

we will further evaluate the individual tasks of the evaluation scenario and the contribution of the pilot interface.

IV. Evaluation

The pilot interface was evaluated based on a DRC-inspired scenario targeting the removal of blocking objects and the turning of an industrial valve. Fig. 7 depicts the complete scenario setup and introduces the labels **T1** - **T5** that will be used throughout this section, referring to the individual objects in the scenario. Note that the affordance extraction pipeline has to be configured before being applicable to a novel scenario, especially regarding segmentation parameters. We do not approach the problem of automatic point cloud segmentation, although sophisticated methods in this area would be applicable. To avoid the problem of segmentation parameterization in this context, the pilot is given the choice between predefined configuration setups for the perceptual pipeline, corresponding to the individual tasks of the evaluation scenario. Sections IV-B and IV-D will show examples where the pilot is able to compensate perceptual shortcomings, i.e. wrongly perceived primitives.

A. Bridging ArmarX and WALK-MAN

The affordance-based pilot interface proposed in this work is implemented in the robot software environment ArmarX [11], which as a framework is robot agnostic. For a robot to be accessible via ArmarX, we need to provide a kinematic robot model and a set of low-level components that connect to the underlying hardware in a standardized manner. Once these components are implemented, existing ArmarX-based skills and user interfaces can be applied to the robot without fundamental porting effort. Fig. 8 illustrates this concept. In [30], ArmarX was integrated with the robot software environment *YARP* [21] in order to connect with the humanoid robot *iCub*. Based on this work, we integrated ArmarX with WALK-MAN. In the following, we will discuss the individual experiments carried out with the pilot interface on the humanoid robot WALK-MAN.

B. Experiment I: The Pipe

In the first experiment, the pilot interface is used for removing the long pipe **T1** (see Fig. 7). The perceptual pipeline successfully identifies the pipe as a distinct primitive and offers the pilot the option to grasp it either unimanually or bimanually. The object is also assigned a *lift* affordance

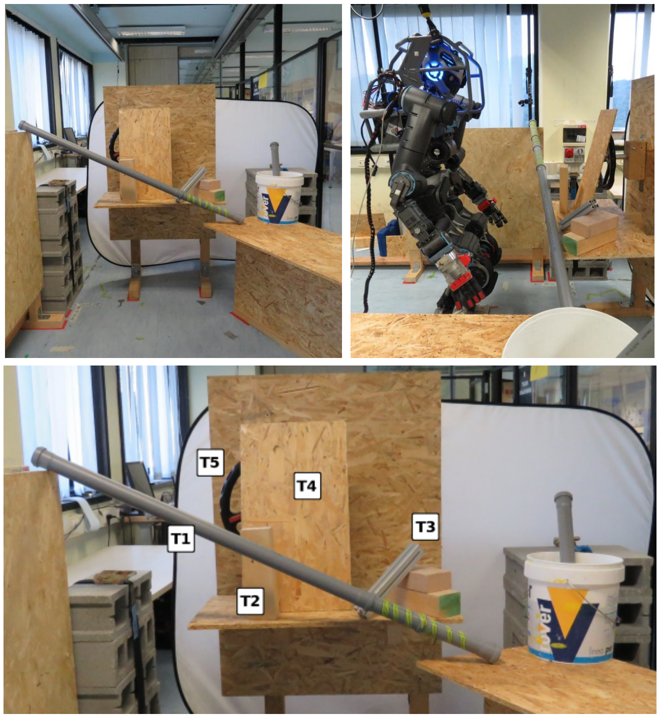


Fig. 7: The experimental setup consists of several objects blocking the robot's access to an industrial valve (**T5**). The objects are a large, horizontal pipe (**T1**), a wooden and a metallic block (**T2** and **T3**) and a wooden board (**T4**).

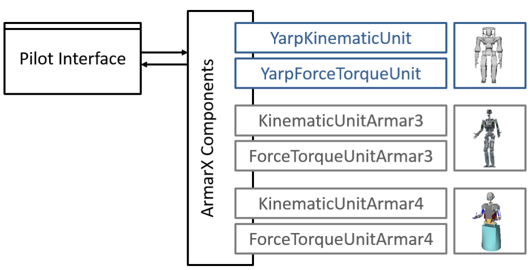


Fig. 8: Bridging ArmarX and YARP for accessing WALK-MAN via ArmarX. A set of low-level components with standardized interfaces need to be implemented, for example a *KinematicUnit* for access to the individual robot joints (for obtaining sensor values and setting control modes and targets) or a *ForceTorqueUnit* for access to available Force/Torque sensors.

hypothesis, which is related to a number of OACs. One of these OACs (*remove*) attempts to lift the object and subsequently moves and drops it in order to remove it from its disturbing position. Other OACs could implement different behavior at this point and would be offered to the pilot in the same way. See Fig. 9 for a visualization of the pilot side of this experiment and Fig. 11 for photos of the corresponding OAC execution with WALK-MAN. The video attachment enclosed with this publication shows a recording of the pilot interface and the subsequent action execution for this experiment. Fig. 9-b shows, that the pipe **T1** was misleadingly extracted as a planar primitive instead of a

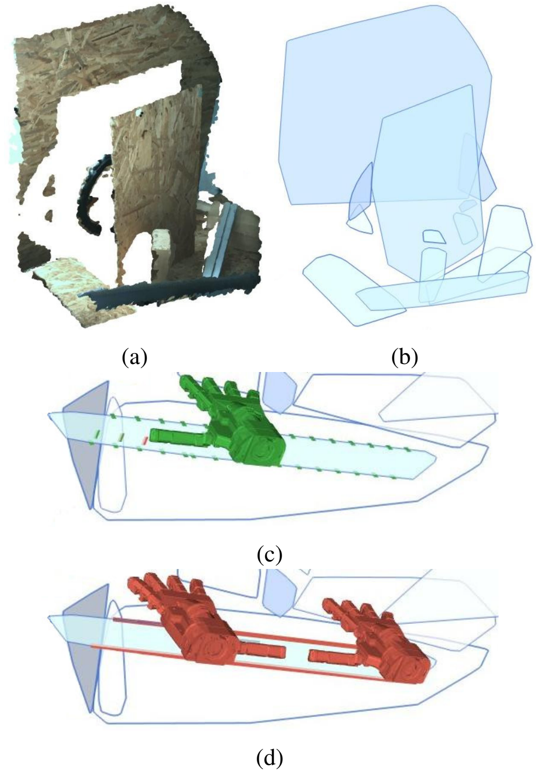
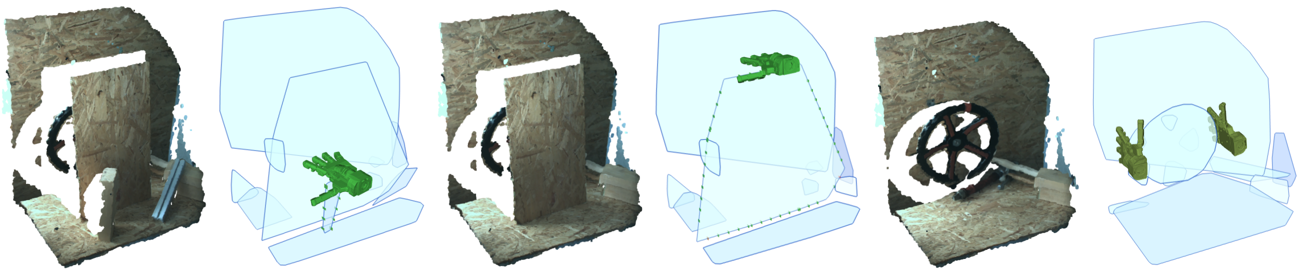


Fig. 9: Visualization of the perceptual process for the first experiment, i.e. the removal of the pipe **T1** (see Fig. 7). The visualizations shown are: (a) the raw point cloud obtained by stereo vision, (b) the extracted primitives in a combined view with the point cloud, (c) the proposed end-effector pose for a unimanual prismatic grasp of the pipe and (d) the proposed end-effector pose for a bimanual prismatic grasp of the pipe.

cylinder. This example shows that the affordance extraction system is flexible enough to account for those perceptual inaccuracies and is still able to offer reasonable affordances and end-effector poses to the pilot. The bimanual prismatic grasp affordance (see Fig. 9-d) receives a low certainty value from the affordance extraction system (red color). This happens due to an internal heuristic in the affordance extraction process that favors roughly horizontal primitive orientations. Details are found in [4]. As can be seen in Fig. 7, the pipe is positioned with a significant incline, making bimanual operation harder.

C. Experiment II: The Blocks

In the second experiment, we commanded the robot to remove the two small block-like objects **T2** and **T3** from the scene. As we will discuss in Section V, the perceptual pipeline is currently not able to recognize compositions of primitives. Hence, it will extract an independent planar primitive for each visible side of the blocks. In this case, the pilot can apply prismatic grasping to the slim sides of the blocks and subsequently remove the objects. See Fig. 10 (a) and (b) for a visualization of the pilot side of this experiment and Fig. 11 for photos of the corresponding OAC execution.



(a) (b) (c) (d) (e) (f)

Fig. 10: Point clouds, extracted primitives and proposed end-effector poses for the different evaluation experiments, i.e. (from left to right) the removal of the two blocks **T2** and **T3**, the removal of the wooden board **T4** blocking the valve and turning the valve **T5**.

D. Experiment III: The Wooden Board

In the third experiment, we commanded the robot to remove the large, wooden board **T4** in front of the valve. The board is correctly detected as a planar primitive, although the segmentation algorithms failed to properly extract its lower bound. Hence, the primitive appears larger than its actual size. The pilot is able to recognize this error and move the proposed end-effector pose for prismatic grasping towards the top side of the board which is properly reflected in the primitive. See Fig. 10 (c) and (d) for a visualization of the pilot side of this experiment and Fig. 11 for photos of the corresponding OAC execution.

E. Experiment IV: The Valve

In the fourth example, after clearing the area around the valve, we commanded the robot to bimanually turn the valve. Here we resembled an experiment on WALK-MAN, that we previously performed on ARMAR-III [4]. The valve is recognized as a planar primitive with a circular shape. Based on this information, the pilot is offered a *bimanual turn* affordance $\Theta_{\text{Bi-Tn}}$ with appropriate end-effector poses for bimanual prismatic grasping. See Fig. 10 (e) and (f) for a visualization of the pilot side of this experiment and Fig. 11 for photos of the corresponding OAC execution.

V. Conclusion and Future Work

Based on our previous work, we proposed an affordance-based pilot interface for humanoid robots. In contrast to existing template-based approaches we aim at unknown environments where environmental objects are not known beforehand. The pilot sees a simplified representation of the environment and is able to control the robot on an abstract level, by choosing among proposed actions and action parameterizations. If the automatic proposal of end-effector poses for OAC parameterization fails, the pilot can intervene and manually adapt the parameterization. The functionality of the affordance extraction process and the pilot interface have been demonstrated in various cluttered scenes on the humanoid robots WALK-MAN and ARMAR-III. The pilot interface presented in this work can be regarded as a proof of concept implementation, for which we see various areas of improvement:

- For recognizing more abstract affordances, the pipeline needs to be able to identify compositions of primitives that form a common structure, e.g. a sequence of horizontal planes forming a staircase, or a combination of perpendicular planes forming a box.
- For supporting sophisticated affordances and OACs, the pilot interface needs to allow the pilot to configure OAC parameters other than end-effector types and poses. For example, in the case of a valve-turning OAC, the degree of turning should be configurable by the pilot.
- We are working on feedback mechanisms that reflect the results of executed OACs and the success rates associated with them (see M in Eq. 2) in affordance certainty functions.
- The proposed approach could be combined with sophisticated methods for the identification of semantic grasp affordances, e.g. for the handling of tools [31], [32].

References

[1] P. Kaiser, N. Vahrenkamp, F. Schültje, J. Borràs, and T. Asfour, “Extraction of Whole-Body Affordances for Loco-Manipulation Tasks,” *International Journal of Humanoid Robotics*, vol. 12, no. 3, 2015.

[2] P. Kaiser, M. Grotz, E. E. Aksoy, M. Do, N. Vahrenkamp, and T. Asfour, “Validation of Whole-Body Loco-Manipulation Affordances for Pushability and Liftability,” in *IEEE-RAS International Conference on Humanoid Robots*, 2015.

[3] J. J. Gibson, *The Ecological Approach to Visual Perception*. 1978.

[4] P. Kaiser, E. E. Aksoy, M. Grotz, and T. Asfour, “Towards a hierarchy of whole-body loco-manipulation affordances,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016.

[5] N. G. Tsagarakis, D. G. Caldwell, A. Bicchi, F. Negrello, M. Garabini, W. Choi, L. Baccelliere, V. Loc, J. Noorden, M. Catalano, M. Ferrati, L. Muratore, A. Margan, L. Natale, E. Mingo, H. Dallali, J. Malzahn, A. Settimi, A. Rocchi, V. Varricchio, L. Pallottino, C. Pavan, A. Ajoudani, J. Lee, P. Kryczka, and D. Kanoulas, “WALK-MAN: A High Performance Humanoid Platform for Realistic Environments,” *Journal of Field Robotics*, 2016.

[6] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, “Armar-iii: An integrated humanoid platform for sensory-motor control,” in *IEEE-RAS International Conference on Humanoid Robots*, pp. 169–175, 2006.

[7] N. E. Sian, K. Yokoi, S. Kajita, F. Kanehiro, and K. Tanie, “Whole body teleoperation of a humanoid robot – development of a simple master device using joysticks –,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 2569–2574, 2002.

[8] R. O’Flaherty, P. Vieira, M. X. Grey, P. Oh, A. Bobick, M. Egerstedt, and M. Stilman, “Humanoid Robot Teleoperation for Tasks with Power Tools,” in *IEEE International Conference on Technologies for Practical Robot Applications*, pp. 1–6, 2013.

T1

T2

T3

T4

T5

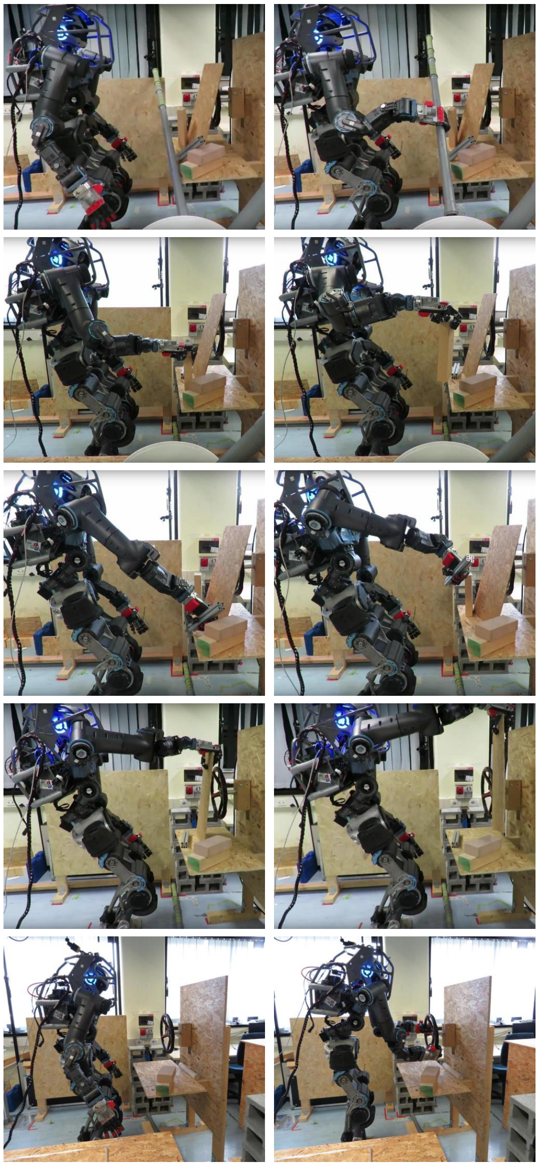


Fig. 11: Photos of OAC executions with WALK-MAN commanded through the pilot interface. The experiments shown are: removal of the pipe (**T1**, first row), removal of the wooden block (**T2**, second row), removal of the metal block (**T3**, third row), removal of the wooden board (**T4**, fourth row) and bimanual turning of the valve (**T5**, fifth row).

[9] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Matarić, "Motion capture from inertial sensing for untethered humanoid teleoperation," in *IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 547–565, 2004.

[10] K. Hambuchen, W. Bluethmann, M. Goza, R. Ambrose, K. Rabe, and M. Allan, "Supervising remote humanoids across intermediate time delay," in *IEEE-RAS International Conference on Humanoid Robots*, pp. 246–251, 2006.

[11] N. Vahrenkamp, M. Wächter, M. Kröhnert, K. Welke, and T. Asfour, "The robot software framework armarx," *Information Technology*, vol. 57, no. 2, pp. 99–111, 2015.

[12] A. Romay, S. Kohlbrecher, D. C. Conner, and O. von Stryk, "Achieving Versatile Manipulation Tasks with Unknown Objects by Supervised Humanoid Robots based on Object Templates," in *IEEE-RAS International Conference on Humanoid Robots*, pp. 249–255, 2015.

[13] M. Fallon, S. Kuindersma, S. Karumanchi, M. Antone, T. Schneider, H. Dai, C. Pérez D'Arpino, R. Deits, M. DiCicco, D. Fourie, T. Koolen, P. Marion, M. Posa, A. Valenzuela, K.-T. Yu, J. Shah, K. Iagnemma, R. Tedrake, and S. Teller, "An Architecture for Online Affordance-based Perception and Whole-body Planning," *Journal of Field Robotics*, vol. 32, no. 2, pp. 229–254, 2015.

[14] S. Nakaoka, M. Morisawa, R. Cisneros, T. Sakaguchi, S. Kajita, K. Kaneko, and F. Kanehiro, "Task Sequencer Integrated into a Teleoperation Interface for Biped Humanoid Robots," in *IEEE-RAS International Conference on Humanoid Robots*, pp. 895–900, 2015.

[15] S. McGill, S.-J. Yi, and D. D. Lee, "Team THOR's Adaptive Autonomy for Disaster Response Humanoids," in *IEEE-RAS International Conference on Humanoid Robots*, pp. 453–460, 2015.

[16] R. Ueda, M. Murooka, Y. Ohara, I. Kumagai, R. Terasawa, Y. Furuta, K. Kojima, T. Karasawa, F. Sugai, S. Iwaishi, S. Nozawa, Y. Kakiuchi, K. Okada, and M. Inaba, "Humanoid Integrated UI System for Supervised Autonomy with Massive Data Visualization over Narrow and Unreliable Network Communication for DRC Competition," in *IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 797–804, 2015.

[17] S. Hart, P. Dinh, and K. Hambuchen, "The Affordance Template ROS Package for Robot Task Programming," in *IEEE International Conference on Robotics and Automation*, pp. 6227–6234, 2015.

[18] R. R. Murphy, "Meta-analysis of Autonomy at the DARPA Robotics Challenge Trials," *Journal of Field Robotics*, vol. 32, no. 2, pp. 189–191, 2015.

[19] P. Birkenkampf, D. Leidner, and C. Borst, "A Knowledge-Driven Shared Autonomy Human-Robot Interface for Tablet Computes," in *IEEE/RAS Int. Conf. on Humanoid Robots*, pp. 152–159, 2014.

[20] C. D. Santina, G. Grioli, M. Catalano, A. Brando, and A. Bicchi, "Dexterity Augmentation on a Synergistic Hand: the Pisa/IIT SoftHand+," in *IEEE-RAS 15th International Conference on Humanoid Robots*, pp. 497–503, 2015.

[21] G. Metta, P. Fitzpatrick, and L. Natale, "YARP: Yet Another Robot Platform," *International Journal of Advanced Robotic Systems*, vol. 3, no. 1, pp. 43–48, 2006.

[22] A. Rocchi, E. M. Hoffman, D. G. Caldwell, and N. G. Tsagarakis, "OpenSoT: a Whole-Body Control Library for the Compliant Humanoid Robot COMAN," in *IEEE International Conference on Robotics and Automation*, pp. 1093–1099, IEEE, 2015.

[23] H. Bruyninckx and P. Soetens, "Open Robot Control Software: OROCOS," 2002.

[24] A. Settimi, C. Pavan, V. Varricchio, M. Ferrati, E. Mingo Hoffman, A. Rocchi, K. Melo, N. G. Tsagarakis, and A. Bicchi, *A Modular Approach for Remote Operation of Humanoid Robots in Search and Rescue Scenarios*, pp. 192–205. Springer International Publishing, 2014.

[25] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To Afford or Not to Afford: A New Formalization of Affordances Toward Affordance-Based Robot Control," *Adaptive Behavior*, vol. 15, no. 4, p. 447, 2007.

[26] L. Jamone, E. Ugur, A. Cangelosi, L. Fadiga, A. Bernardino, J. Piater, and J. Santos-Victor, "Affordances in psychology, neuroscience and robotics: a survey," *IEEE Transactions on Cognitive and Developmental Systems*, 2016.

[27] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, A. Agostini, and R. Dillmann, "Object-action complexes: Grounded abstractions of sensorimotor processes," *Robotics and Autonomous Systems*, vol. 59, pp. 740–757, 2011.

[28] E. Ovchinnikova, M. Wächter, V. Wittenbeck, and T. Asfour, "Multipurpose natural language understanding linked to sensorimotor experience in humanoid robots," in *IEEE/RAS International Conference on Humanoid Robots*, pp. 365–372, 2015.

[29] M. Wächter, S. Ottenhaus, M. Kröhnert, N. Vahrenkamp, and T. Asfour, "The armarx statechart concept: Graphical programming of robot behaviour," *Frontiers - Software Architectures for Humanoid Robotics*, 2016.

[30] A. Paikan, D. Schiebener, M. Wächter, T. Asfour, G. Metta, and L. Natale, "Transferring object grasping knowledge and skill across different robotic platforms," in *International Conference on Advanced Robotics*, pp. 498–503, 2015.

[31] D. Kanoulas, J. Lee, D. G. Caldwell, and N. G. Tsagarakis, "Visual Grasp Affordance Localization in Point Clouds using Curved Contact Patches," *International Journal of Humanoid Robotics*, 2016.

[32] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Detecting Object Affordances with Convolutional Neural Networks," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016.