

# Template-Based Manipulation in Unstructured Environments for Supervised Semi-Autonomous Humanoid Robots

Alberto Romay\*, Stefan Kohlbrecher\*, David C. Conner†, Alexander Stumpf\* and Oskar von Stryk\*

**Abstract**—Humanoid robotic manipulation in unstructured environments is a challenging problem. Limited perception, communications and environmental constraints present challenges that prevent fully autonomous or purely teleoperated robots from reliably interacting with their environment. In order to achieve higher reliability in manipulation we present an approach involving remote human supervision. Strengths from both human operator and humanoid robot are leveraged through a user interface that allows the operator to perceive the remote environment through an aggregated worldmodel based on onboard sensing, while the robot can efficiently receive perceptual and semantic information from the operator. A template based manipulation approach has been successfully applied to the Atlas humanoid robot; we show real world footage of the results obtained in the DARPA Robotics Challenge Trials 2013.

## I. INTRODUCTION

Autonomous and teleoperated manipulation using humanoid robots are still complex problems. While challenges such as object recognition, self-localization and obstacle avoidance have to be tackled for autonomous systems, purely teleoperated robots require dealing with communication constraints such as loss of information and latency as well as providing the proper feedback and situational awareness to the operator.

If we consider a fully autonomous robot navigating and interacting in an unconstrained environment, the capabilities of this robot should include extensive databases of information about possible objects of interest to be found, highly efficient grasping algorithms and the ability to react to unforeseen situations, which are still unsolved problems. On the other hand, if we consider a purely teleoperated robot, the capabilities of the robot and the operator should include near real-time feedback without disruptions in the communications as well as transmission of large amounts of data to the operator. The performance of teleoperated robots also strongly depends on the experience and ability of the human operator to interact with the environment.

Now, as a third option, if we consider a human-robot cooperation that implements the strengths from both, the challenges of either fully autonomous and purely teleoperated robots could be tackled.

### A. Related Work

Fully autonomous robots have the advantage of not needing communications and independence of robot control from

\*Department of Computer Science, TU Darmstadt, Germany  
romay, kohlbrecher, stumpf, stryk@sim.tu-darmstadt.de

†TORC Robotics, VA, USA conner@torcrobotics.com

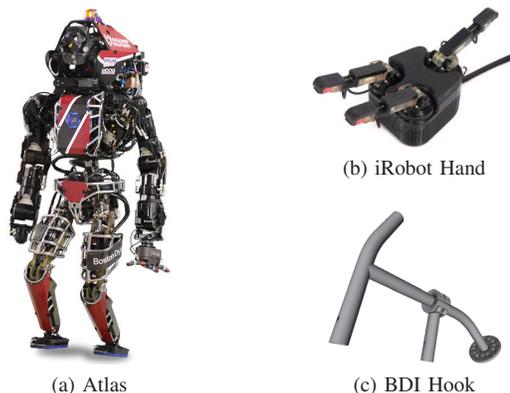


Fig. 1: a) Atlas robot with 28 hydraulically actuated DOF, stands 1.88 m tall and weighs approximately 150 Kg. b) iRobot Hand [17], a 3 fingered robotic hand with 5 DOF and c) a metal hook developed by BDI.

an operator. While a lot of research has been performed for autonomous humanoid robots in structured environments with impressive results [16], [21], fewer results have been obtained for humanoid robots in unstructured environments. Full autonomy presents challenges such as object recognition and mission planning, which given the conditions that can be encountered in unstructured environments, make fully autonomous robots not yet feasible to perform these tasks efficiently.

To try to overcome the challenges of autonomously identifying objects and understanding how these objects can be grasped and used, some researchers have studied the concept of affordances. The term “affordances” was first introduced by the psychologist J.J. Gibson in [6]. An affordance, from the perspective of Psychology, is a term that describes the possible actions that an object offers to an organism in the environment. This concept has been adopted by research fields related to robotics as an approach to define how a robot should behave in order to accomplish a determined task using an object. It is the relationship between an object and how a robot should manipulate this object. The extensive work of Şahin et al. [20] to formalize the term of affordance in the robotics field has been key to the development of robot control approaches [1], [25]. Kruger et al. [11] proposed Object-Action Complexes (OACs), which aim at defining the relationships between objects and actions. OACs is a concept created to formalize this relationship and it has been used to develop control approaches that aim at allowing robots with the possibility to understand how an object

will behave after an action is performed over it. OACs allow autonomous robots to learn and predict the behaviours derived from performing an action over an object as well as to build symbolic representations of continuous sensorimotor experience.

The work of Nagatani et al. [15], [14] showed that the use of teleoperated robots can help obtaining information of hazardous environments. In their project, they use a wired network to overcome communication issues, but this also leads to limited exploration range and risk of entanglement. During the mission of the robot Quince to the Fukushima nuclear plant in 2011, the access to the third floor was blocked by rubble; semi-autonomous manipulation of such unstructured objects could have led to continued exploration.

The approach presented in [5] is similar to ours. The differences are the way in which we define the possible actions to be executed over an object and how we interact with them. In their approach, for example, they developed automatic template fitting algorithms which are faster than human assisted alignment. While in our approach, motion planning happens onboard the robot, they create the motion plans in their control station and send them afterwards. Another difference is the way we approach the term affordance. For example, to use the open-ability and close-ability of a valve, in their approach, the operator manually rotates the template of the valve to create a motion plan, while in our approach, the operator specifies a semantic action such as “open” or “close” using a number like “ $\pm 360$  degrees”.

## B. Overview

Humans can work in a highly abstract, discrete space, having the knowledge and the perception capacity to easily identify and classify objects of interest (by their semantic properties), as well as decision making to accomplish a task. Robots have the capacity to perform calculations on a continuous space estimating objects’ physical properties (e.g. mass and inertia). Mixing the strengths of both, humans can in principle aid a remote semi-autonomous humanoid robot to perform rescue tasks in environments that are dangerous for humans. In order to command a robot through a remote environment, we make use of a graphic user interface or Operator Control Station (OCS), where a human operator can use a 3D visualization of the remote environment through the information provided by the robot’s sensors.

To be able to provide the robot with information about the environment it perceives, we created an approach that incorporates physical and semantic information about the objects of interest into a template that is sent to the robot through the OCS. This object template concept is inspired by the theory of affordances [6] and OACs [11] (see Section I-A). The theory of affordances and OACs define the concept of “object” as an entity of the environment. However, for the purposes of our approach, we created the concept of “object template” as an augmentation of the concept “object” which includes not only physical information such as a 3D mesh, mass and center of mass, but also includes semantic information such as grasp types, potential end effector poses, robot

stand poses and information about how the robot should manipulate the objects (explained in detail in Section II). This additional information is of high relevance because it simplifies remote semi-autonomous robot control by providing knowledge about how the state of the environment is and how to interact with it.

Human supervision of semi-autonomous robots implies the robot needs the capability to autonomously perform parts of manipulation tasks. Humanoid robots are complex due to the high number of degrees of freedom (DOF), making direct joint-based teleoperation infeasible. In order to execute manipulation tasks, robots must provide motion planning capabilities [2] considering collision avoidance [12], as well as geometrical world model information such as 3D point clouds [19] and/or grid maps for locomotion planning. This information also needs to be communicated to the human operator to provide situational awareness, supplemented by additional information such as still images and video streams as needed. However, this remote communication is subject to constraints such as interrupts, bandwidth and latency. To overcome some of these communication constraints, the robot must compress and abstract the sensor data transmitted to the human operator. More details about semi-autonomous manipulation, world modelling and providing situational awareness to a remote human operator will be discussed in Section III.

This paper focuses on the improvement of human-robot cooperation to perform manipulation tasks in complex environments based on object templates, where humanoid robots are remotely commanded by a human operator, taking advantage of the capabilities and strengths of humans and robots. We can summarize our contributions as follow:

- Human supervision of semi-autonomous robots by the proposed approach can significantly increase the efficiency to perform manipulation tasks in unconstrained environments compared to fully autonomous or purely teleoperated robots.
- A concept of object templates is proposed which, inspired by the theory of affordances and object-action complexes, allows a fast communication of information from the operator to the robot, which would be error prone and difficult to obtain if the robot would have to extract it autonomously.
- This template based manipulation approach has been implemented in an operator control station and its effectiveness has been demonstrated to accomplish tasks representative for rescue or recovery missions in a disaster scenario (see Section V).

The announcement of the DARPA Robotics Challenge (DRC) gave us the opportunity to test our developments. We participated in the DRC as Team ViGIR<sup>1</sup>, a cooperation between our research group at the Technische Universität Darmstadt in Germany and other research institutions in the USA. As members of a track B team, the DRC had imposed challenging time constraints on the development;

<sup>1</sup><http://www.teamvigir.org>

we had 8 months to implement our software in a simulation environment to participate in the Virtual Robotics Challenge (VRC) [9] and practically less than three months to implement and test our software in the real Atlas humanoid robot (Fig. 1) developed by Boston Dynamics Inc. (BDI) before participating in the DRC Trials in December 2013 (Section IV). To speed up our developments and in spirit of open source code, we make use of highly advanced open source libraries such as Robot Operation System (ROS) [18], MoveIt! [2], GraspIt! [13] and the Gazebo simulator as complementation for our own software developments. A general description of the complete approach to the DRC including details on OCS, footstep planning, bandwidth-constrained communication and kinematics calibration is presented in [10].

## II. OBJECT TEMPLATES

To be able to provide fast and efficient semantic information of the objects of interest to the robot, we created templates of known objects. Graphically, these templates are visualized as a simplified 3D mesh of the object it represents. They are designed to be a general shape of the real object so they can be used for similar objects (e.g. drills from different brands). These object templates contain additional information about each object such as mass and center of mass (CoM), we also created grasp templates to provide pre-computed potential pre-grasp and final-grasp poses, basic grasp types (e.g. cylindrical, prismatic, spherical), as well as information about possibilities of action over each object. This concept of possibilities of action over an object has been previously studied and researched as Affordances [20] or Object-Action complexes [11] (see Section I-A).

Using object templates, the human operator can aid the robot to identify objects of interest in cluttered sensor data as well as their respective properties. In the OCS, the operator can overlap the 3D mesh of an object template with the visualization of the sensor data corresponding to the object of interest. This way, the robot can use the relative pose of the 3D object template to estimate the real object's pose. Once the robot has an estimate of the real pose of the object, the operator can then iterate through the pre-computed list of grasp templates visualizing the arm configurations prior to performing motions on the real robot (Fig. 2).

### A. Grasp Template Definition

A grasp template contains the necessary information about where and how to grasp an object template. It allows the operator to visualize the arm configuration needed to grasp the object before actually performing a motion with the real robot. Grasp templates are defined by the tuple:

$$g = (H, E, N, S, P_p, P_f),$$

where:

- $H = \{1, 0\}$  :  $LeftHand = 1, RightHand = 0$ ,
- $E = \{cylindrical, prismatic, spherical\}$  defines the type of grasp to be used,

- $N$  is the vector of fingers joint values where the fingers make contact with the object,
- $S \in \mathbb{R}^2$  defines the desired 2D position of the robot pelvis relative to the template,
- pose  $P_p \in \mathbb{R}^3 \times \mathbb{SO}(3)$  defines the position and quaternion orientation of the hand for the pre-grasp, and
- pose  $P_f \in \mathbb{R}^3 \times \mathbb{SO}(3)$  defines the position and quaternion orientation of the hand for the final-grasp.

Several grasps templates are created offline for each object template using the GraspIt! simulator [13]. Different grasps  $E$  with their corresponding final finger joints  $N$  are created for each object and tested a priori in simulation. Pre-grasps poses  $P_p$  are potential hand poses to place the hand in a position near the object (we designed these poses to be around 10 cm away from the object) to reduce the risk of collision (Fig. 2a) while reaching the final-grasps poses  $P_f$ , which are the final poses that the hand needs to reach before closing the fingers around the object (Fig. 2b).

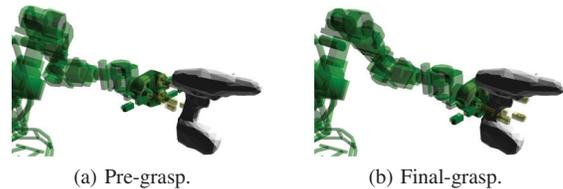


Fig. 2: Using a “ghost robot” the pre-grasp and final-grasp poses (here shown for the drill template) can be visualized prior to perform an arm motion on the real robot.

To visualize the pre-grasps and final-grasps for each object template, a 3D transparent hand or “ghost hand” is projected in the pose relative to the template. That way, the human operator can choose the location of the hand for a particular task (Fig. 3). The pose  $S$  of the robot pelvis relative to the object is selected in a way that allows the end effector to reach the object with high manipulability [24] (Fig. 5b).

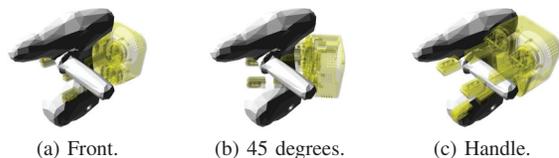


Fig. 3: Final grasps for the drill template.

### B. Object Template Definition

Once we have a list of grasp templates for each object we can now create object templates defined by the tuple:

$$x = (I, T, M, C, G, U),$$

where:

- $I \in \mathbb{N}$  is the ID number of the object of interest,
- $T \in \mathbb{N}$  is the type of template (e.g. tools, debris, hose),

- $M \in \mathbb{R}$  is the estimated mass of the object,
- $C \in \mathbb{R}^3$  is the estimated CoM of the object,
- $G$  is a list of potential grasp templates  $g$ ,
- $U = \{T_x, T_y, T_z, R_x, R_y, R_z\} \in \{0, 1\}^6$  is a six-dimensional vector (3D translation and rotation) that defines if an action is possible over a dimension.

Inspired by the theory of affordances from J.J. Gibson [6] we created a vector  $U$  to define which actions are possible to be performed over an object. We constrained these actions to translations and rotations in a defined (but not unique) frame of reference in a template. That way, we can command the robot through our OCS to perform actions to a template, for example, the valve can be turned, the drill can be pushed and the door can be opened by turning the handle and pushing or pulling. These constrained path motions of the robot’s hand are described in Section III-E. Table I shows examples of the possible actions for the drill, door and hose templates.

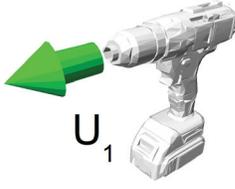
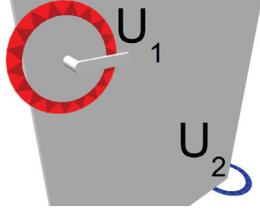
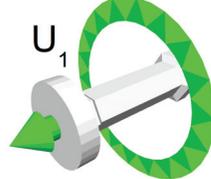
Drill Template 	$U_1 = \{1, 0, 0, 0, 0, 0\}$  The drill action possibility is a translation along the $X$ axis (green arrow).
Door Template 	$U_1 = \{0, 0, 0, 0, 1, 0\}$ $U_2 = \{0, 0, 0, 0, 0, 1\}$  The door action possibilities are to rotate around the $Y$ axis (red ring) in $U_1$ and rotate around the $Z$ axis (blue ring) in $U_2$ .
Hose Template 	$U_1 = \{1, 0, 0, 1, 0, 0\}$  The hose action possibility is a translation and a rotation around the $X$ axis (green arrow and ring).

TABLE I: Possible actions over object templates.

### III. OBJECT TEMPLATE MANIPULATION

To be able to perform manipulation tasks using object templates, information of the environments as well as from the robot state needs to be gathered.

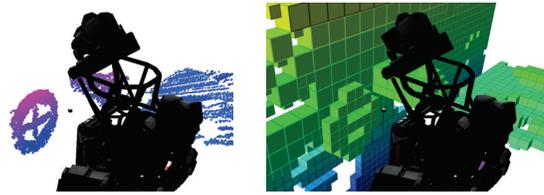
#### A. World Model

A 3D model of the environment is generated by aggregating sensor data from LIDAR and cameras. A pose estimate of the robot is obtained by the IMU on its pelvis, and we keep track of different coordinate frames in order to fully reconstruct the pointclouds requested relative to different fixed frames. Using robot pose estimate, internal

joint sensing and external sensors, we generate a world model that can also be visualized in the OCS (Fig. 4). This model is used for multiple applications such as visualizing all joint states, self filtering from sensor data and collision avoidance.

#### B. Planning

Once the world model is obtained, the sensor data is processed and simplified to generate efficient motion plans. To avoid collisions with the environment [12], the robot creates a 3D Octomap [7] representation of the world model (Fig. 4b). For locomotion planning, 2D grid map slices from regions of interest are created and used to generate a collision free footstep plan [8], [23] (Fig. 10c).



(a) World and Robot model.

(b) 3D Octomap.

Fig. 4: World model of a valve scenario with robot model and the 3D Octomap for planning.

#### C. Providing Situational Awareness to the Operator

The robot has access to full resolution sensor data, however, given all the possible constraints that the communication link might be subject to, this information needs to be compressed in order to provide the human operator with situational awareness. Although only joint states and IMU information is sent periodically to the operator, additional sensor data like 3D pointclouds, 2D images and video are provided on request. To reduce the amount of bandwidth used, sensor data is down-sampled and cropped according to the operator’s request (Fig. 8d).

#### D. Pipeline with templates

In this section we will describe a use case of our template manipulation approach (Fig. 5).

1) *Sense*: The human operator requests sensor data from the robot’s environment to gather situational awareness.

2) *Plan*: After identifying the sensor data that corresponds to the real object, the human operator overlaps an object template (Fig. 5a). The ghost robot then moves to the pose  $S$  of the template, and places the end effector in the final-pose  $P_f$  (Fig. 5b).

3) *Walk*: The human operator commands the robot to walk to the stand pose  $S$  for which a footstep plan is generated (Fig. 5c). The robot executes this plan.

4) *Grasp*: Once the robot reaches the stand pose  $S$ , the human operator commands the robot to grasp the object and the robot generates a motion plan. The pre-grasp  $P_p$  pose, the final-grasp  $P_f$  pose and the grasp  $N$  are then executed.

5) *Use*: The human operator can then command the robot to generate a manipulation plan for the object based on the possible actions  $U$ .

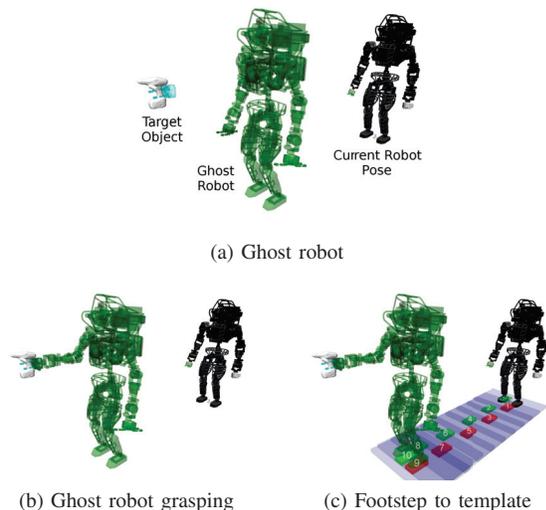


Fig. 5: Ghost robot grasp preview with a footstep plan to reach the object [23].

### E. Cartesian and Circular Path planning

The vector  $U$  defined in each template is used to create motions that are constrained to follow a Cartesian path between the initial and final end effector's pose. Waypoints are generated based on linear interpolation between initial and final poses. By using spherical linear interpolation (Slerp) [22] orientations for the end effector's goal pose can be different from the start end effector's orientation. More complex constrained motions such as circular motion are generated by concatenating multiple short linearly interpolated Cartesian paths. These constrained motions can also be designed to maintain the end effector's orientation (Fig. 6). This video<sup>2</sup> shows the operator commanding the robot to close a valve in a simulation environment.

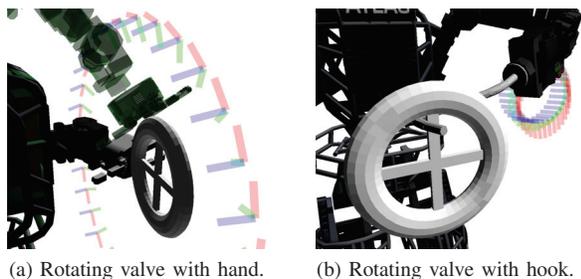


Fig. 6: Circular path plan to turn a valve 360 degrees. In (a) the hand rotates around changing its orientation, while in (b) the hook rotates around keeping its orientation. Interpolated poses are shown for the last joint of the arm.

## IV. EXPERIMENTS AND RESULTS

The DRC Trials 2013 were held in Homestead, FL, USA. Research groups from different countries participated in a

series of tasks to demonstrate robot capabilities for rescue missions. These tasks considered robot capabilities such as mobility and manipulation in disaster environments like:

- Walk through rough terrain,
- Climb up a ladder,
- Remove debris blocking a doorway,
- Open three different types of doors,
- Break through a wall using a cutting tool,
- Attach a fire-hose to a wye,
- Close three different types of valves and
- Drive a car.

Seven of the eight tasks required grasping and manipulation. Each of the tasks consisted of three defined checkpoints. A point was given for each accomplished checkpoint and in case all checkpoints were accomplished without an intervention (robot failure which required a restart) an extra bonus point was given. The tasks required different amounts of mobility and manipulation and for the purposes and scope of this paper we will describe the Hose task and the Valve task because they contain good examples where we applied our template based manipulation approach. A comprehensive description of all eight task results is presented in [10].

### A. Hose Task

In the Hose task, the robot needed to walk to a reel and pick up a fire hose, then walk with the fire hose towards a wye and attach it by turning the nozzle (Fig. 7a). The first point in this task was obtained when the robot crossed the yellow line on the floor while carrying the hose. The second point was given when the hose came in physical contact with the wye and the third point was obtained for attaching the hose. Our approach to accomplish this task was first to divide it in three subtasks: pick up the hose, touch the wye with the hose and attach it. The figures shown in this section contain screenshots from the OCS, either a top-down view of the environment or a 3D view. The figures also contain images of the real scenario, obtained from different cameras located in the walls of the task (Fig. 7a).

1) *Pick up the hose*: Following the pipeline described in Section III-D we began the task by acquiring sensor data information from the environment. The secondary operator requests a pointcloud of the reel and inserts a hose template aligning it to the 3D data belonging to the real hose (Fig. 7b). Then the primary operator requests a footstep plan to a position relative to the hose template where the robot can easily grasp the hose as shown in Fig. 7c and Fig. 7d. Once the robot is standing in front of the hose, the primary operator requests a grasp to pre-grasp pose of the ghost hand (Fig. 8a and Fig. 8b), then the robot moves the hand to the final-pose and executes the grasp (Fig. 8c and Fig. 8d).

After grasping the hose the primary operator commands the robot to move one meter to the right based on an environment map previously requested by the secondary operator (Fig. 9).

2) *Touch the wye with the hose*: For this subtask we applied the same pipeline again. The secondary operator requests a pointcloud of the wye (Fig. 10a) and inserts the

<sup>2</sup><https://www.youtube.com/watch?v=wKFJO-Zkj4c>

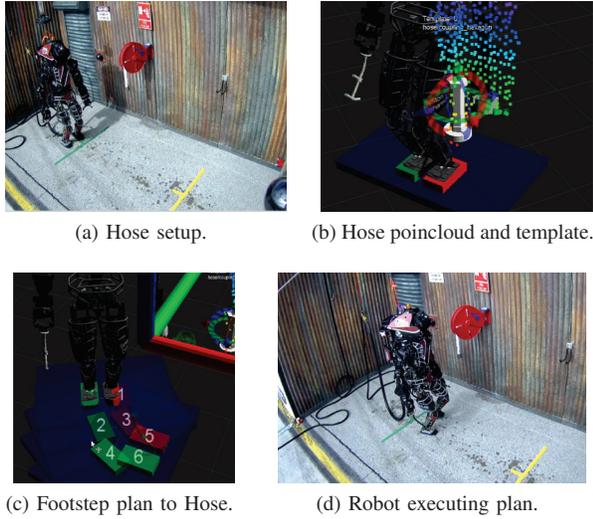


Fig. 7: Request to walk to the hose. (a) Robot start position, hose reel and wye. (b) Reel and hose pointcloud with the template of the hose. (c) Footstep plan visualization to the hose in the OCS before walking. (d) Robot following the footstep plan to the hose.

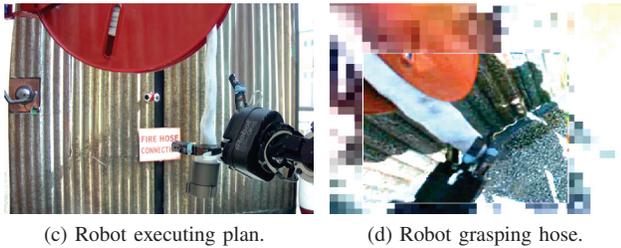
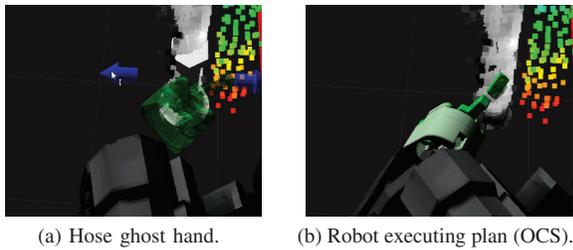


Fig. 8: Request to grasp hose. (a) Hose ghost hand visualization for the operator to verify. (b) and (c) Automatic motion of the robot to place the hand in final-pose. (d) Operator verifies the grasp through robot's camera.

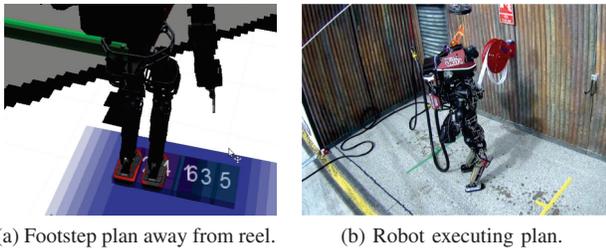


Fig. 9: Request to walk away from reel. (a) Footstep plan with lateral right steps. (b) Robot walking with the hose.

wye template, aligning it to the 3D data belonging to the real wye (Fig. 10b). Then the primary operator requests a footstep plan to a position relative to the wye template where the robot can easily touch the wye with the hose (Fig. 10c and Fig. 10d). Once the robot is standing in front of the wye, the primary operator request the robot to move the hand to the pre-grasp pose of the ghost hand (Fig. 11a), which the robot executes (Fig. 11b and Fig. 11c). Then the robot moves the arm to the final-grasp pose which makes the hose to come in physical contact with the wye (Fig. 11d).

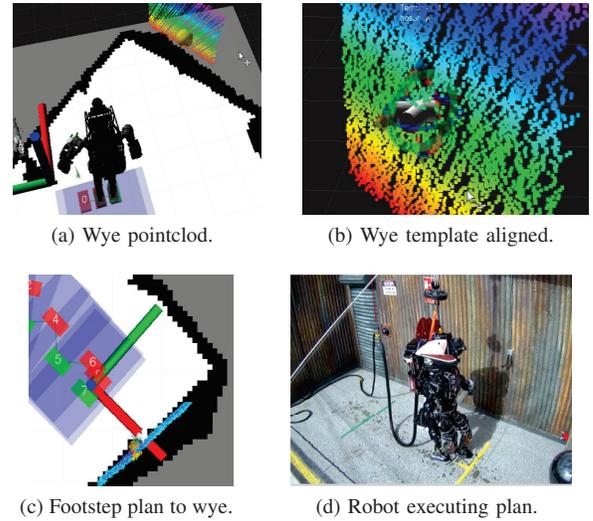


Fig. 10: Request to walk to the wye.

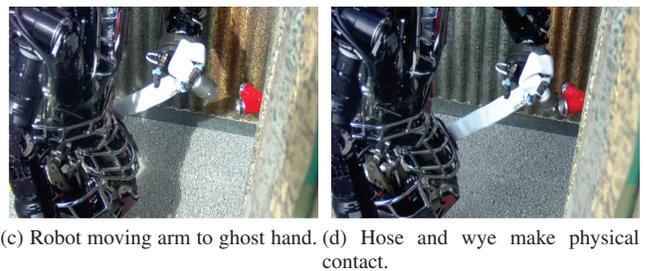
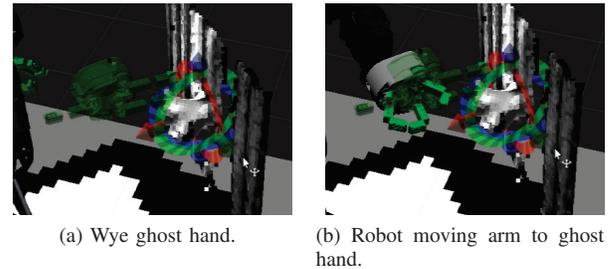


Fig. 11: Robot commanded to bring the hand near the wye and then touch it.

3) *Attach the hose:* At this point we have successfully applied our template based approach to align the fire hose to the wye and the only missing thing is turning the nozzle

to engage the hose. Given the extremely small size of the nozzle bumps used to turn it (around  $0.25 \text{ cm}^3$ ), this subtask was not feasible to solve using our approach. Teleoperation was used instead, however, this was a fine manipulation task which required high precision and even the hose was correctly located (Fig. 12a) and the nozzle was turned 180 degrees (Fig. 12b), the threads of the wye and the hose did not engage, and the hose fell after releasing it.

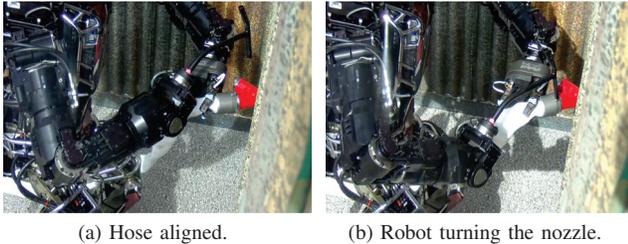


Fig. 12: Attach hose and turn nozzle (teleoperated).

### B. Valve Task

For accomplishing the Valve task, the robot needed to close three different valves (Fig. 13a). We followed the same pipeline as in the Hose Task in order to command the robot to place the hook inside the valves as shown in Fig. 13b).

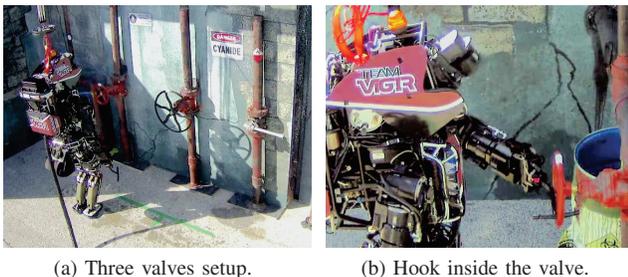


Fig. 13: Valve task.

Once the robot placed the hook inside the valve and the valve template was aligned, the primary operator selects the option to close the valve from a menu in the OCS simply by indicating the amount of degrees. As shown in Fig. 14, the robot starts moving the arm in a circular path as described in Section III-E. This circular motion plan is always relative to the axis of rotation of the valve template.

### C. Results

To compare our results, we analysed the performance of other teams using the Atlas robot during the DRC Trials. Table II shows the timetables of the activities performed during the Hose Task day one [3] and day two [4].

These results show that we were the fastest to perform manipulation tasks. We were able to walk with the hose through the yellow line in the floor within 8 minutes, touch the wye at time 10:10, align it with the wye at 18:00 and start turning it at time 22:20. From these results we can see

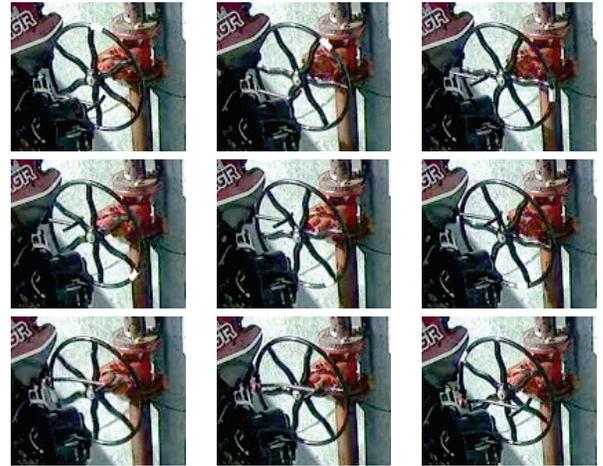


Fig. 14: Robot autonomously closing a real valve.

that there were only two teams able to turn the nozzle of the hose, and in most of the other manipulation activities we were faster than other teams.

## V. CONCLUSIONS AND FUTURE WORK

We have presented a humanoid robot manipulation approach based on templates of objects of interest. The proposed approach enables human-robot cooperation to perform manipulation tasks in a more efficient way compared to pure teleoperated and full autonomous robot approaches. The use of object templates allows a human operator to efficiently send semantic commands to a remote humanoid robot, such as “Walk to this waypoint”, “Grasp this object” or “Turn this object” so a mission can be planned on the fly. As shown in Section IV our template based manipulation approach is very useful for grasping and manipulating various objects. Some limitations of our approach are that small objects that require fine manipulation are still a challenging problem and also that the operator has to invest time to manually align templates to sensor data. As shown in Table II we were the fastest Atlas team to obtain two points in the Hose Task and the fastest to turn the nozzle from the only two teams that tried. We can see how, even though we placed 9<sup>th</sup> overall in the Trials, our template manipulation approach gave better results than the 2<sup>nd</sup> place results in the Hose Task.

Future work will focus on developing automatic template fitting algorithms to increase the speed of our approach. We are also researching automatic grasp planners to be able to create new grasps on the fly. Our current definition of templates only allows to define the possibilities of action as constrained motion paths, but we would like to explore options for adding information about expected forces needed for these motions. An analysis of some (anonymous) DRC teams results has been done in [26] which we are analysing to find opportunities for improvements in our approach.

## ACKNOWLEDGMENT

The authors would like to thank all members of Team ViGIR, their contribution and support enabled the realization

Activity	Team ViGIR	IHMC	MIT	TRACLabs	WRECS	Trooper	HKU
Task Start Time (min)	00:00	00:00	00:00	00:00	00:00	00:00	00:00
Stand in front of Hose	00:40	01:15	02:15	01:00	01:00	03:25	03:40
Picked Hose	04:00	08:40	03:53	06:20	03:10	14:40	-
First Point (Crossed Line)	<b>08:00</b>	11:50	09:30	08:45	07:55	29:30	-
Stand in front of Wye	08:30	13:25	11:00	09:15	08:18	-	-
Second Point (Touched Wye)	<b>10:10</b>	15:20	15:25	16:25	11:28	-	-
Hose alignment	18:00	16:20	22:35	29:00	-	-	-
Turned Nozzle	<b>22:20</b>	23:00	-	-	-	-	-
Points	2	2	2	2	2	1	0
Place overall DRC Trials	9 <sup>th</sup>	2 <sup>nd</sup>	4 <sup>th</sup>	6 <sup>th</sup>	6 <sup>th</sup>	8 <sup>th</sup>	10 <sup>th</sup>

TABLE II: Hose Task timetable for all Atlas teams [3], [4].

of this work; we would especially like to thank Alex Goins and Dr. Ravi Balasubramanian from Oregon State University for assistance getting the grasping software working and testing, and Felipe Bacim and Dr. Doug A. Bowman of Virginia Tech for their work in the operator interface design and development. This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under contract FA8750-12-C-0337, by the Deutscher Akademischer Austauschdienst (DAAD Germany), and by the Secretaría de Educación Pública (SEP México).

#### REFERENCES

- [1] A. Bierbaum and M. Rambow, "Grasp affordances from multi-fingered tactile exploration using dynamic potential fields," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*. IEEE, 2009, pp. 168–174.
- [2] S. Chitta, I. Sucan, and S. Cousins, "Moveit!" *IEEE Robotics Automation Magazine*, vol. 19, no. 1, pp. 18–19, 2012.
- [3] DARPA Robotics Challenge Trials, "DRC Trials Blue Hose Task Day 1," <https://www.youtube.com/watch?v=n3aGCe0VDLU>, 2013, accessed: 2014-10-11.
- [4] —, "DRC Trials Blue Hose Task Day 2," <https://www.youtube.com/watch?v=B35A7-JDIdI>, 2013, accessed: 2014-10-11.
- [5] M. Fallon, S. Kuindersma, S. Karumanchi, M. Antone, T. Schneider, H. Dai, C. Perez D'Arpino, R. Deits, M. DiCicco, D. Fourie, *et al.*, "An architecture for online affordance-based perception and whole-body planning." 2014. [Online]. Available: <http://hdl.handle.net/1721.1/85690>
- [6] J. J. Gibson, "The theory of affordances," in *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, R. Shaw and E. J. Bransford, Eds., Hilldale, USA, 1977, pp. 67–82.
- [7] A. Hornung, K. M. Wurm, M. Bennis, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013, software available at <http://octomap.github.com>. [Online]. Available: <http://octomap.github.com>
- [8] A. Hornung, A. Dornbush, M. Likhachev, and M. Bennis, "Anytime search-based footstep planning with suboptimality bounds," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*. IEEE, 2012, pp. 674–679.
- [9] S. Kohlbrecher, D. C. Conner, A. Romay, F. Bacim, D. A. Bowman, and O. von Stryk, "Overview of Team ViGIR's approach to the Virtual Robotics Challenge," in *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 1–2.
- [10] S. Kohlbrecher, A. Romay, A. Stumpf, A. Gupta, O. von Stryk, F. Bacim, D. A. Bowman, A. Goins, R. Balasubramanian, and D. C. Conner, "Human-robot teaming for rescue missions: Team ViGIR's approach to the 2013 DARPA Robotics Challenge Trials," 2014, accepted for Journal of Field Robotics.
- [11] N. Krüger, C. W. Geib, J. H. Piater, R. P. A. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrcen, A. Agostini, and R. Dillmann, "Object-action complexes: Grounded abstractions of sensory-motor processes." *Robotics and Autonomous Systems*, vol. 59, no. 10, pp. 740–757, 2011. [Online]. Available: <http://dblp.uni-trier.de/db/journals/ras/ras59.html#KrugerGPPSWUAKOAD11>
- [12] A. Leeper, K. Hsiao, M. Ciocarlie, I. Sucan, and K. Salisbury, "Methods for collision-free arm teleoperation in clutter using constraints from 3d sensor data," in *IEEE Intl. Conf. on Humanoid Robots*, Atlanta, GA, 10/2013 2013.
- [13] A. Miller and P. K. Allen, "Graspit!: A Versatile Simulator for Robotic Grasping," *IEEE Robotics Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.
- [14] K. Nagatani, S. Kiribayashi, Y. Okada, K. Otake, K. Yoshida, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, M. Fukushima, *et al.*, "Emergency response to the nuclear accident at the fukushima daiichi nuclear power plants using mobile rescue robots," *Journal of Field Robotics*, vol. 30, no. 1, pp. 44–63, 2013.
- [15] K. Nagatani, S. Kiribayashi, Y. Okada, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, and Y. Hada, "Redesign of rescue mobile robot quince," in *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 13–18.
- [16] M. Nieuwenhuisen, J. Stückler, A. Berner, R. Klein, and S. Behnke, "Shape-primitive based object recognition and grasping," in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*. VDE, 2012, pp. 1–5.
- [17] L. U. Odhner, L. P. Jentoft, M. R. Claffee, N. Corson, Y. Tenzer, R. R. Ma, M. Buehler, R. Kohout, R. D. Howe, and A. M. Dollar, "A compliant, underactuated hand for robust manipulation," *International Journal of Robotics Research*, 2013.
- [18] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, vol. 3, no. 3.2, 2009.
- [19] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–4.
- [20] E. Şahin, M. Çakmak, M. R. Doğan, 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, pp. 447–472, 2007.
- [21] D. Schiebener, J. Schill, and T. Asfour, "Discovery, segmentation and reactive grasping of unknown objects." in *Humanoids*, 2012, pp. 71–77.
- [22] K. Shoemake, "Animating rotation with quaternion curves," *ACM SIGGRAPH computer graphics*, vol. 19, no. 3, pp. 245–254, 1985.
- [23] A. Stumpf, S. Kohlbrecher, D. C. Conner, and O. von Stryk, "Supervised footstep planning for humanoid robots in rough terrain tasks using a black box walking controller," in *Humanoid Robots (Humanoids), 2014. 14th IEEE-RAS International Conference on*. IEEE, 2014.
- [24] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Robot placement based on reachability inversion," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1970–1975.
- [25] K. Welke, J. Issac, D. Schiebener, T. Asfour, and R. Dillmann, "Autonomous acquisition of visual multi-view object representations for object recognition on a humanoid robot," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2012–2019.
- [26] H. Yanco, A. Norton, W. Ober, D. Shane, A. Skinner, and J. Vice, "Analysis of human-robot interaction at the DARPA Robotics Challenge Trials." 2014, submitted to Journal of Field Robotics.