**2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**
**October 25-29, 2020, Las Vegas, NV, USA (Virtual)**

# Building Plannable Representations with Mixed Reality

Eric Rosen, Nishanth Kumar, Nakul Gopalan, Daniel Ullman, George Konidaris, Stefanie Tellex

Department of Computer Science, Brown University

*Abstract*— **We propose Action-Oriented Semantic Maps (AOSMs), a representation that enables a robot to acquire object manipulation behaviors and semantic information about the environment from a human teacher with a Mixed Reality Head-Mounted Display (MR-HMD). AOSMs are a representation that captures both: a) high-level object manipulation actions in an object class's local frame, and b) semantic representations of objects in the robot's global map that are grounded for navigation. Humans can use a MR-HMD to teach the agent the information necessary for planning object manipulation and navigation actions by interacting with virtual 3D meshes overlaid on the physical workspace. We demonstrate that our system enables users to quickly and accurately teach a robot the knowledge required to autonomously plan and execute three household tasks: picking up a bottle and throwing it in the trash, closing a sink faucet, and flipping a light switch off.**

## I. INTRODUCTION

A long-term goal of robotics is designing robots intelligent enough to enter a person's home and perform daily chores for them. This requires the robot to learn specific behaviors and semantic information that can only be acquired after entering the home and interacting with the humans living there. For example, there may be a trinket that the robot has never encountered before, and the owner might want to instruct the robot on how to handle the item (i.e., object manipulation information), as well as directly specify where the item should be kept (i.e., navigation information). To approach this problem, one must consider two sub-problems: a) the agent's representation of **object manipulation actions** and **semantic information about the environment**, and b) the method with which an agent can **learn this knowledge from a teacher**.

Semantic maps provide a representation sufficient for navigating an environment [1, 2], but map information alone is insufficient for enabling object manipulation. Conversely, there are knowledge bases that store requisite object manipulation information [3–7], but do not help with navigation or grasping in novel orientations. Previous studies [8–10] have shown that Mixed Reality (MR) interfaces are effective for specifying navigation commands and programming ego-centric robot behaviors. However, none of these works have demonstrated the use of MR interfaces for teaching high-level object manipulation actions, and semantic information of the environment.

Our contribution is a system that enables humans to teach robots both object manipulation actions—in a local object frame of reference—and b) semantic information about objects in a global map. We use a Mixed Reality Head Mounted Display (MR-HMD) to enable humans to teach a robot a plannable representation of their environment. By *plannable*, we mean structured representations that are searchable with AI planning tools [11]. By *teach*, we mean having the human explicitly provide information necessary for instantiating our representation. Our representation, the Action-Oriented Semantic Map (AOSM), enables robots to perform complex object manipulation tasks that require navigation around an environment. To test our system for building AOSMs, three novice humans used our MR interface to teach a robot an AOSM, allowing the robot to autonomously plan to navigate to a bottle, pick it up, and throw it out. In addition, we report the quantitative results of two expert users who demonstrated the power of learning AOSMs via MR by also teaching a robot to autonomously plan to flip a light switch off (Figure 1) and manipulate a sink faucet to the closed position. To the best of our knowledge, this is the first work that presents a learnable representation for planning manipulation and navigation tasks on a robot via an MR interface.

## II. RELATED WORK

Numerous previous works have combined low-level metric maps with high-level topological and semantic information [1, 12–17], but with a focus on navigational tasks. Various works have made semantic map representations that use a hybrid of metric, topological, and conceptual representations [1], and incorporated human input to improve and teach these representations for the purpose of navigation [12, 15, 16]. Most notably, Pronobis and Jensfelt [12]'s semantic map representation has place appearance and geometry, object information, topology, human input, segmentation, conceptual maps, uncertain concepts, inferred properties, and autonomously acquired concepts. However, Pronobis and Jensfelt [12] do not learn object manipulation requisites, such as grasp points, termination sets, and motor policy representations, and only focus on information necessary for effective localization and navigation.

Previous works have learned object representations that do contain information that is used for object manipulation planning [3–7], but do not consider learning semantic map representations of their environment in the process. Object-Action-Complexes (OACs) [3, 4], which consider objects and action representations to be intertwined by capturing interactions between objects and associated actions, allow the agent to acquire object knowledge about the world through predicting changes in the world via agent interaction. While OACs provide a symbolic representation of sensor-

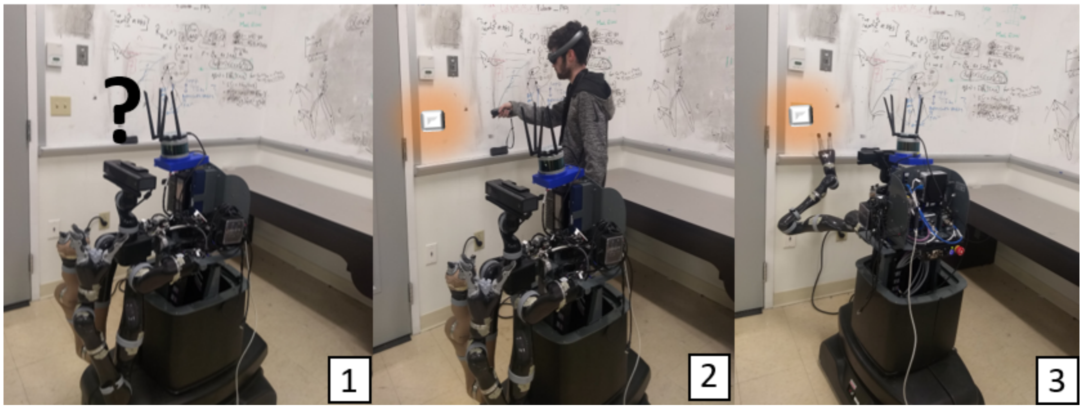

Fig. 1: Our MR system being used to generate an AOSM so the robot can flip a light switch. **(1)**: Initially, the robot does not know the location of the light switch, how to grasp it, nor how to turn it off. **(2)**: A human using MR teaches the robot the object's global pose $\Gamma$, the "grasp" attribute, and the initiation and termination pose for the "turn off" action (highlighted in orange).**(3)**: The robot is now able to autonomously plan to navigate to the lights, motion plan to grasp the light switch, and execute the policy to flip it off.

motor experience for objects, they do not have sufficient information about the environment to generate maps for the use of navigation. Beetz et al. [5] present an impressive knowledge-base, KNOWROB2, which incorporates components like perception, learning, and reasoning to achieve complicated manipulation tasks like making a pizza. While KNOWROB2 is able to learn what robot poses in a map of the environment are useful for actions like suitable grasps, it requires access to an "inner world" of the environment with symbolically-annotated objects with the map, and does not address how to learn such a detailed and accurate semantic map representation of the environment. More importantly, KNOWROB2 does not represent actions in local object frames, which is crucial for leveraging the MR interface and teacher input. Our proposed method helps the agent acquire both the semantic map, and plannable representations for manipulation from a human teacher by instead leveraging the underlying virtual environment model of a MR-HMD and representing actions within object frames.

MR-HMDs show great promise for facilitating human-robot interaction, and have been used for communicating robot motion trajectories [9, 18, 19] and specifying robot commands [10]. Beyond their improvements to speed, accuracy, and mental workload over baselines [9], MR-HMDs also enable the human to share the same space as the robot and interact with a virtual environment instead of having to interact with the real, physical robot [8]. While projector-based approaches are also a powerful tool for facilitating human-robot interaction [20], they require structured environments and are unable to highlight free 6D space because they must project onto a surface, which is limiting in the case where a human must teach spatial attributes (like a grasp pose) for planning.

While some previous work has used MR-HMDs to have robots learn from humans, it has only focused on simple pick and place tasks, and not on using the MR interface to learn requisite information needed for complex object manipulation and navigation [8, 21]. Gadre et al. [8] designed an MR interface to enable end-users to program robot motions via waypoint specification for the purpose of pick and place, and Krupke et al. [21] designed a MR interface for a similar task, but instead manipulated virtual items in the workspace to specify place locations. While these works demonstrate the capability of learning with MR, they focus on how such an interface compares to other modalities (like 2D monitor interfaces). We address the problem of acquiring semantic maps and high-level action requisites that is needed for navigation and complex object manipulation behaviors on a robot from a human with a MR-HMD.

## III. ACTION-ORIENTED SEMANTIC MAPS

We first formalize AOSMs by describing the object classes, object instances, and object manipulation actions it contains, which are all defined with a local frame. Next, we illustrate the grounding of an object's semantic information to a global frame of reference. Lastly, we describe our method of using MR to build an AOSM from a human trainer.

### *A. Defining Action-Oriented Semantic Maps*

An Action-Oriented Semantic Map is a tuple $AOSM = \langle C, O, M, A \rangle$, where $C$ is a set of object classes; $O$ is a set of objects instantiated from $C$, where we define instantiation as assigning all attributes of a class to values representing a real-world object; $M$ is a 2D occupancy grid of the environment; and $A$ is a set of high-level object-specific actions parameterized over objects in $O$.

Each high-level action $a \in A$ is akin to an option [22]. Given an object from $o \in O$ and a high-level action $a$, a "policy", an "initiation set", and a "termination set" for the option is specified. In the next subsection we will describe using MR to acquire each of these components in detail.

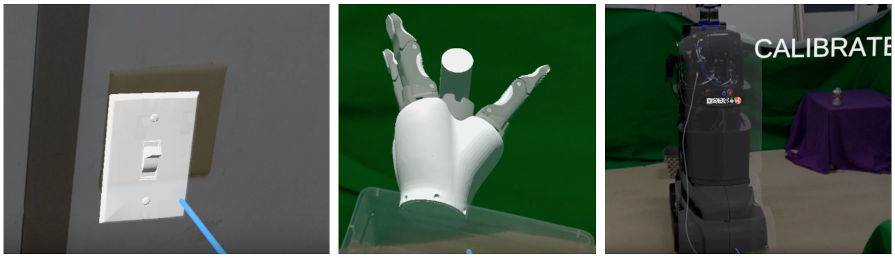

Fig. 2: The perspective of a user with our MR interface (all visualizations here are from the actual interface). **Left:** A closeup image of a user grounding the global pose $\Gamma$ of the light switch using our MR interface. **Middle:** A user specifying the terminating pose for the "throw away" action of the bottle object, with the annotated grasp pose visualized as a white robot gripper. **Right:** An image of the virtual robot overlaid on top of the real robot while calibrating the MR-HMD's map with the robot's map $M$. The text "CALIBRATE" indicates to the user what information they are specifying.

Each class within $C$ is constructed with a set of attributes $\alpha$, a 6D local frame $\Lambda$, a global pose $\Gamma$, and a kinematic mesh model $\tau$. The 6D (position and orientation) local frame $\Lambda$ is necessary to define spatial attributes for high-level actions with respect to an object, regardless of the global pose $\Gamma$. The model $\tau$ is defined in the local frame with respect to $\Lambda$.

The 3D kinematic mesh model of each object class, $\tau$, is specified with respect to $\Lambda$. The purpose of the 3D mesh is twofold. Firstly, it provides an interface for the teachers to manipulate and specify the local coordinate frame of an object so that the skill specification is intuitive. Secondly, it allows a manipulable interface to the object in MR allowing the trainer to visualize and manipulate a virtual object, which is the typical mode of interaction with an object in MR.

Each class has a set of attributes $\alpha$, akin to class attributes in Object-Oriented Markov Decision Processes [23]. Attributes are used to represent information required for planning object manipulation behaviors. Spatial attributes, like "grasp", which defines how the object should be grasped with respect to $\tau$, are specified with respect to $\Lambda$. In our experiments, the only attribute we have for our classes is "grasp", but other complex domains require more attributes for planning.

Once objects are instantiated, they have a global pose $\Gamma$ in the map, and the agent knows where the object is and can navigate to it. Moreover, a high-level action $a$ is defined with respect to the local frame $\Lambda$ of the object class $c$. Specifically, the policy, initiation set and termination set of a high-level action $a$ are all defined in the object class's local frame. This allows transfer of learned high-level actions to different objects within the same class and to different poses in the global frame, enabling the robot to reproduce and generalize the learned skill later when executing a plan.

Whenever an object is instantiated, $\Gamma$ is grounded to $M$, and $\tau$ is rendered by the MR-HMD based on $\Gamma$. $\Gamma$ is the pose of the local frame's origin with respect to $M$'s origin. The purpose of the global pose $\Gamma$ is so that information defined with the local frame $\Lambda$ for an object is now grounded within the map $M$, enabling the robot to know where in the environment it should navigate to in order to perform object manipulation behaviors. The purpose of rendering $\tau$ in MR is so that the teacher can specify $\Gamma$ by dragging the virtual object model, and directly view whether $\Gamma$ is correctly specified (i.e: if the virtual $\tau$ is overlaid on top of the real object).

### B. Instantiating AOSMs with Mixed Reality

In order to ground the poses of the virtual items to our semantic map, the map maintained by the MR-HMD must be linked to the robot map $M$. MR-HMDs already have a built-in capability to make a 3D mesh model of the environment for mapping, which is used for localization. However, there is no inherent link between the MR-HMD's map and the robot map $M$, which is required to use MR to specify an object's global pose $\Gamma$. To resolve this issue, a static transform that defines how to convert global poses in the virtual environment maintained by the MR-HMD to poses in the robot's map must first be defined (Figure 2). Our method of performing this calibration using MR is explained in Section IV.

After calibrating the MR-HMD and the robot map, the user can teach the robot object manipulation and semantic information of the environment (as described in Section IV-B). The user is presented with a list of object classes $C$ from the AOSM. When the user selects a class, a virtual representation of the object's mesh $\tau$ is visualized in front of the user as a 3D mesh (Figure 2), and an interaction process is initiated, where the user supplies each of the necessary attribute values within the object's local frame $\Lambda$. For example, in case of the "grasp" attribute, the user is presented with a visualization of the object's mesh $\tau$ along with a virtual model of the robot's end effector (Figure 2). The user is then able to pose the virtual end effector to grasp the virtual object mesh $\tau$. Users are able to supply manipulation information using the high-level actions for an object by filling in the parameters. The users first select an object to add an high-level action to, and then manipulate a virtual representation of the object's mesh $\tau$ into the desired initiation and termination poses. Because $\tau$ is an articulated 3D mesh model, users can specify the initiation and termination poses by selecting a link with the controller, and then manipulate it with their controller to the desired pose. For the purposes of our MR interface implementation, these initiation and termination poses were in terms of the mobile-manipulator's end effector so that our system could check when these poses were reached. This process allows users to not have to specify any low-level manipulation control such as environment-specific grasp operations.

Because there are several design choices for the MR interface that can be made based on the desired task, we selected several household tasks and conducted an iterative design study to understand what factors were important for enabling novice humans to teach a robot an AOSM. This design process allowed us to include features that were not initially considered by the expert designers, but were desired by the novice users.

## IV. ITERATIVE DESIGN STUDY

We conducted an iterative design study with two expert users (two of the project researchers) and three novice users in order to design and improve our MR interface, as well as demonstrate the capabilities of AOSMs.[1]

### A. Study Task

To demonstrate that an AOSM can be built by a human using MR, we selected several household tasks for a mobile manipulator to perform, which we represent within what we term the "Household AOSM". We chose three different chores: throwing away bottles, turning off light switches, and closing sink faucets. Our test environment is shown in Figure 3.

Each element of our Household AOSM ($AOSM = \langle C, O, M, A\rangle$) is defined as follows:

- $C$: a list of three object classes: bottle (a drinking container with no kinematic articulation), faucet (a sink faucet, which has a revolute joint connected to a sink base, which could be closed), and light switch (which has a revolute joint connected to the wall). Each of the classes have a 6D local frame $\Lambda$, a global pose $\Gamma$, and a kinematic mesh model $\tau$. In order to keep the AOSM as simple as possible, we only encoded one attribute: "grasp", which represents a 6D pose in the class frame $\Lambda$ that indicated how to grasp the object for manipulation.
- $O$: a list of the instantiated objects from the list of classes. In our experimental space, we had one bottle, one light switch, and one sink faucet. Rather than requiring users to build $\tau$ from scratch, we supplied various primitive shapes and predefined object models for the user to choose from to represent the objects, which is reasonable considering there are many existing object models freely available to be downloaded [24]. Therefore, when instantiating objects, users were responsible for defining the "grasp" attribute needed for the high-level action manipulation actions, as well as the global pose $\Gamma$ of the object within the map $M$ which is needed for navigation (Figure 2).
- $M$: a 2D occupancy grid $M$ that represents the experimental space (Figure 3). The map is updated with new semantic information when an object $o$ is instantiated and its global pose $\Gamma$ is grounded in the map. It is this underlying map that enables the robot to autonomously plan navigation around the environment.
- $A$: For our demonstration, we paired one high-level action with each object to represent the three chores. However, it should be be noted our framework is flexible enough to allow an arbitrary number of high-level actions to be defined throughout the interaction by the user. Our actions are as follows:
  1) For the bottle class objects, the high-level action "throw away" was meant to pick up a bottle and move it to a trash can in a fixed spot (Figure 2).
  2) For the light switch class, a "turn off" high-level action was meant to flip the switch to the off position from the on position.
  3) For the sink faucet class, a "close faucet" high-level action was meant to close the faucet.

  Users were responsible for using our MR interface to define the initiation and termination poses of these actions, while the policy attached $\pi$ was implemented using an existing motion planner to move the robot's end effector to the grasp pose with respect to the initiation pose, and then compute and execute a motion to manipulate the object to the termination pose. The policy was first planned within the local frame $\Lambda$, and then transformed into the global map frame based on $\Gamma$, enabling the robot to move its end effector to the necessary locations in the map to manipulate the object. We can also use Dynamical Movement Primitives [25] as a policy within the local frame $\Lambda$.

Our study was implemented on a Kinova Movo with a single 7 DoF arm. Movo is equipped with the capability to make a 2D occupancy map of its environment using a LIDAR sensor, as well as localize and navigate to specified poses. When planning any of the object manipulation actions, the robot would autonomously move its base between 0.8 and 1.25 meters behind the object's global pose $\Gamma$, depending on what the "grasp" attribute and global pose $\Gamma$ of the object was, enabling the agent to execute the local policy and manipulate the object into its termination pose from the initiation pose (Figure 1). While this range of approach distances was chosen by hand for the purposes of completing our specified chores, they could in practice be specified by the user via the MR interface. For all of these motion behaviors, we use the motion and path planning stack that is included with the Movo robot. By supplying our metric map $M$ to the path planning stack, we are able to autonomously navigate the robot to specific points while avoiding occupied space.

### B. Mixed Reality Interface

The two most commercially-available MR-HMDs are the Microsoft HoloLens and the MagicLeap. We have previously

[1] A video can be found at https://youtu.be/-09b250TTe8

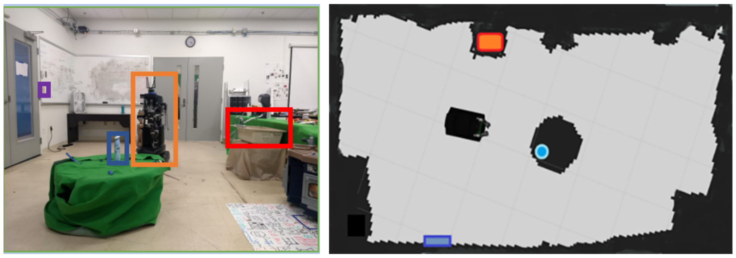

Fig. 3: Images from our Household AOSM. **Left:** One perspective image with our three classes $C$ being instantiated with objects $O$ (light switch (purple), bottle (blue), sink (red)) and robot (orange). **Right:** The 2D occupancy grid $M$ in our Household AOSM (Colored shapes and robot added to the map for visualization purposes).

used the Microsoft HoloLens for facilitating human-robot interactions [9], but chose to use the Magic Leap for this work because it provides higher precision head-pose estimation. However, the following work can be applied to any MR-HMD system. Our codebase for the MR interface is publicly available.[2]

We used Unity, a 3D game engine, to develop the virtual environment for the MR interface, by developing a scene that maintains virtual objects, and deploy it to the Magic Leap. By connecting the Magic Leap to a ROS network, we are able to share information between the MR interface and a ROS-enabled robot. (A more detailed description of how this can be done may be found in Whitney et al. [26]). Crucially, our system is developed such that no objects in the Unity scene need to be pre-instantiated; the user is able to construct the scene completely at runtime via our MR interface.

The Unity-ROS interface allows the Unity scene to output information on the ROS network to communicate to the robot, or listen to information from the robot to update the virtual scene. In general, MR interfaces enable users to see visualizations of 3D meshes overlaid on top of the physical workspace, as well as interact with these visualizations using controllers or hand gestures. We leverage MR to enable users to instantiate virtual representations of the objects from a set of classes using an MR menu, supply attribute and high-level action information needed for object manipulation by interacting with the model $\tau$ of objects in order to specify initiation and termination poses, and ground objects to the map (i.e: specify $\Gamma$) for the purpose of navigation by dragging the virtual objects over their real-world counterparts (Figure 1).

To define the static transform that is needed to convert Unity poses to ROS poses, we enable users to drag a virtual model of the robot over the real one to align them together (Fig 2), similarly to how they would ground the global pose $\Gamma$ of an object. After the user drags the virtual robot over the real one, we save the transformation from the virtual robot pose to the real robot pose as the static transform from Unity to ROS poses. With this transform, we now have a way to use a MR-HMD to ground poses of objects in the robot's map. More information on pose transformation between MR-HMDs and robot maps can be found in Whitney et al. [26].

### *C. Rapid Iterative Testing and Evaluation*

We took a Rapid Iterative Testing and Evaluation (RITE) [27] approach to quickly identify and fix issues with the system.

For the purpose of the iterative design study, we limited the Household AOSM by removing the light switch and sink faucet from the Household AOSM, and only focused on the bottle object. Users in our study were instructed to specify the "grasp" attribute for the bottle class, the initiation and termination pose of the "throw away" action, and the global pose $\Gamma$ of the object in the map. The expert users then completed the full Household AOSM by also handling the sink faucet and light switch.

We built an initial interface for the system, and tested and iterated on the design of the interface. We tested the initial system with two expert users (two of the project researchers), iterated on the design, and then tested and iterated with three novice users, who used our interface until they successfully performed the task. We then tested the final system with the expert users.

*1) System V0:* We started with an initial design for the MR interface, system V0, that was derived from previous MR interfaces we have used with robotic systems [9]. The interface allows users to drag virtual representations over objects in the real world that they want to interact with, as described in the Section IV-B. However, we noticed that users have slight calibration issues with hand gestures (i.e., it is hard to accurately capture hand gestures), such that we decided to use a hand controller instead to circumvent this calibration issue. We drew inspiration from the MagicLeap's toy app which uses the hand controller to orient objects in front of the controller. Thus, our initial design improved on our previous interfaces by introducing a hand controller to replace gesture in order to attempt to address user issues with positioning virtual representations.

We then tested system V0 with the expert users. We quickly found that the expert users would sometimes unknowingly misalign the virtual representations over the real-world objects. For example, after specifying the global pose for a specific bottle, the user would walk around the room to specify other attribute and action information; however, after physically walking in the room, and thus changing perspective, the user would notice that global pose of the

[2] https://github.com/h2r/ActionOrientedSemanticMaps

object appeared misaligned with the real-world object. We implemented an intervention (i.e., edit) function for the sequence of human actions for an item, such that the MR interface would permit users to respecify and edit information in the AOSM. We also noticed that scenes would sometimes become cluttered with specifications for multiple items; consequently, we implemented a color scheme for objects to make differentiation between virtual objects easier.

*2) System V1:* We tested system V1 with the first novice user. The major observation from the user concerned the sensitivity of the hand controller, which the user found to be overly sensitive to touch and thus difficult to use to precisely position the virtual representations. We reduced the sensitivity of the controller for the subsequent version.

*3) System V2:* Feedback from the second novice user centered on a desire to know what action they were specifying for the robot at any given time, as they sometimes lost their place in the sequence while adding states. We addressed this issue for the subsequent version by implementing a text display in the virtual workspace that identifies whether they were specifying action information, object pose/attribute information or calibration information (Figure 2).

*4) System V3:* The third novice user tested system V3, and did not have any major issues with using the system.

We therefore proceeded to test system V3 with the original expert users. The experienced users were able to use this final version of the system to complete more complex cleanup tasks, such as turning off sinks and turning off lights.

### D. Overall Impressions of System

The interviews with the three novice users revealed that, overall, they liked the system and found the system intuitive when they used it.

One notable consideration revealed during user testing concerns sensitivity of the hand controller; users varied in how sensitive they wanted the hand controller to be in response to their input. The first novice user found the hand controller too sensitive (prompting a reduction in sensitivity); the second novice user did not report any issues with sensitivity; the third novice user found it not sensitive enough.

Ultimately, the insight from novice user expectations of the system helped guide the design of the final system. The two expert users tested the final system with complex tasks of flipping light switches and turning off faucets.

As predicted, a major problem of the MR interface was the decalibration due to drift. Over time, users would see the virtual objects drift away from their calibrated poses because the MR-HMD was not able to accurately localize itself within a large space with a constantly moving user, making their groundings inaccurate for the robot. To resolve this, multiple interventions to edit specified information was required. Although allowing users to readjust the transform between Unity and ROS made this issue less pressing, users reported that it was cumbersome to do this repeatably. Therefore, high-precision pose tracking is crucial for using MR to specify semantic information about the robot's environment. Another option is to incorporate autonomous perception modules beyond SLAM into the MR interface, such as object detection and pose estimation, which can leverage the user-specified information to enable the object's pose estimate to be robust to decalibration due to drift between the robot and the MR-HMD. The human-specified information can be used in conjunction with iterative computer vision algorithms, like ICP for pose registration [28], which are are sensitive to initial starting points and would benefit from human input.

## V. RESULTS

In order to evaluate our system, we demonstrated that our final MR interface enables an expert user to build the full Household AOSM to sufficiently perform all three high-level behaviors: navigating to a bottle and throwing it away, navigating to a light switch and turn it off, and navigating to a sink faucet to close it. For each object, users were tasked with specifying an object's $\Gamma$, "grasp" attribute, and the initiation and termination pose for the associated action (as discussed in Section IV). Once the users trained the robot with this information, the robot was able to plan with the Household AOSM. For planning, the agent autonomously performs a multi-step plan of a) moving to a position near the object's $\Gamma$ (as described in Section IV), b) grasping the object based on the "grasp" attribute and initiation pose, and c) manipulating the object into its termination pose (Figure 1). Whenever the agent fails to execute the plan, we enable the user to intervene (i.e., edit) any specified information.

To quantify our evaluation, we recorded both the total time it took to teach the high-level action, specify the global pose of the instantiated object, and have the robot autonomously plan to execute the behavior. In addition to the total time, we also record the number of interventions required until a successful plan is executed.

There is no fair baseline comparison to our method because we are the first work to present a representation that has both semantic and planning information that is learnable via MR. Comparing against direct teleoperation or kinesthetic teaching in the real workspace is not a valid baseline because there is no way to specify the position and orientation of all the links in an object by controlling the robot's arm, which is needed for specifying the initiation and termination pose of an action. A 2D visual interface that uses our metric map $M$ is also not a valid baseline because it does not provide any geometric information about the location of the objects, only geometric information of obstacles slightly above floor height, and therefore can not be used to label object pose information. Making a 3D static map of the environment and visualizing it on a 2D monitor is also not a valid baseline because user intervention requires a dynamically updated model of the room to respecify information, which a static map does not provide. Continuously mapping a large 3D dynamical scene with on-board robot sensor data is not a fair comparison because it requires the user to move the robot to acquire desired view points, introducing a conflating factor of robot control that is not encountered with the MR interface. A projector-based augmented reality interface is also not a feasible comparison because it does not provide a method for manipulating or visualizing 3D kinematic mesh models, which is necessary for defining our high-level actions.

For the bottle task, our expert user took 31 seconds, and had 0 interventions. For the light switch task, the expert user took 91 seconds, and had 4 interventions. For the sink faucet task, the expert user took 45 seconds and 3 interventions. Note that the total times include all of the interventions (i.e: the timer was not stopped between each intervention). Therefore, the light switch and sink faucet task have longer reported times due to the number of interventions needed to complete the task, but the average intervention time for the light switch task was 22.75 seconds, and 15 seconds for the sink faucet. It took less than 2 minutes to complete each of our tasks.

## VI. CONCLUSION

We present a solution to enable users to teach robots both high-level actions for object manipulation and semantic map representations for navigation via an MR interface. We introduced Action-Oriented Semantic Maps (AOSMs), a plannable representation which can enable a human to teach a robot information needed for object manipulation and navigation through MR. To demonstrate that humans can build AOSMs to plan for complex object manipulation tasks, we showed that novice and expert users can program a mobile manipulator to perform three tasks: picking up a bottle and throwing it in the trash, closing a sink faucet, and flipping a light switch.

While our approach has shown significant promise, there are limitations to the MR-HMD interface and there exists room for improvements. For example, our iterative design study revealed that it is crucial to maintain accurate localization of the MR-HMD within the map, which is more difficult to guarantee in highly-unstructured and dynamic domains. We hope to improve the MR interface by enabling physical rotation of the hand controller to rotate the object in the virtual space, rather than using the touch pad on the hand controller. We also hope to include a calibration phase for the MR interface's sensitivity to address individual variation across users. Finally, a natural extension would be to generate object classifiers from the teacher's annotations to enable autonomous perception of objects in the room, as well as using MR with AOSMs to conduct human-in-the-loop reinforcement learning to improve object manipulation policies.

## REFERENCES

[1] M. R. Walter, S. Hemachandra, B. Homberg, S. Tellex, and S. Teller, "Learning semantic maps from natural language descriptions," in *Robotics: Science and Systems*, 2013.

[2] M. Hanheide, M. Göbelbecker, G. S. Horn, A. Pronobis, K. Sjöö, A. Aydemir, P. Jensfelt, C. Gretton, R. Dearden, M. Janicek, H. Zender, G.-J. Kruijff, N. Hawes, and J. L. Wyatt, "Robot task planning and explanation in open and uncertain worlds," *Artificial Intelligence*, vol. 247, pp. 119–150, Jun. 2017. [Online]. Available: http://www.pronobis.pro/publications/hanheide2017ai

[3] F. Wörgötter, A. Agostini, N. Krüger, N. Shylo, and B. Porr, "Cognitive agentsa procedural perspective relying on the predictability of object-action-complexes (oacs)," *Robotics and Autonomous Systems*, vol. 57, no. 4, pp. 420–432, 2009.

[4] N. Krügera, C. Geibb, J. Piaterc, R. Petrickb, M. Steedmanb, F. Wörgötterd, A. Udee, T. Asfourf, D. Krafta, D. Omrcene *et al.*, "Object-action complexes: Grounded abstractions of sensorimotor processes."

[5] M. Beetz, D. Beßler, A. Haidu, M. Pomarlan, A. K. Bozcuoğlu, and G. Bartels, "Know rob 2.0a 2nd generation knowledge processing framework for cognition-enabled robotic agents," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 512–519.

[6] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Transferring skills to humanoid robots by extracting semantic representations from observations of human activities," *Artificial Intelligence*, vol. 247, pp. 95–118, 2017.

[7] D. Paulius and Y. Sun, "A survey of knowledge representation and retrieval for learning in service robotics," *arXiv preprint arXiv:1807.02192*, 2018.

[8] S. Y. Gadre, E. Rosen, G. Chien, E. Phillips, S. Tellex, and G. Konidaris, "End-user robot programming using mixed reality," in *Proceedings of the IEEE International Conference on Robotics and Automation (in press)*. IEEE, 2019.

[9] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex, "Communicating robot arm motion intent through mixed reality head-mounted displays," in *International Symposium on Robotics Research*, 2017.

[10] B. Huang, D. Bayazit, D. Ullman, N. Gopalan, and S. Tellex, "Flight, camera action! Using natural language and mixed reality to control a drone," *ICRA*, 2019.

[11] T. Kurutach, A. Tamar, G. Yang, S. J. Russell, and P. Abbeel, "Learning plannable representations with causal infogan," in *Advances in Neural Information Processing Systems*, 2018, pp. 8733–8744.

[12] A. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 3515–3522.

[13] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86–103, 2015.

[14] A. Nüchter and J. Hertzberg, "Towards semantic maps for mobile robots," *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 915–926, 2008.

[15] E. Bastianelli, D. Bloisi, R. Capobianco, G. Gemignani, L. Iocchi, and D. Nardi, "Knowledge representation for robots through human-robot interaction," *arXiv preprint arXiv:1307.7351*, 2013.

[16] T. Spexard, S. Li, B. Wrede, J. Fritsch, G. Sagerer, O. Booij, Z. Zivkovic, B. Terwijn, and B. Krose, "Biron, where are you? Enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 934–940.

[17] M. Hanheide, M. Göbelbecker, G. S. Horn, A. Pronobis, K. Sjöö, A. Aydemir, P. Jensfelt, C. Gretton, R. Dearden, M. Janicek *et al.*, "Robot task planning and explanation in open and uncertain worlds," *Artificial Intelligence*, vol. 247, pp. 119–150, 2017.

[18] M. Walker, H. Hedayati, J. Lee, and D. Szafir, "Communicating robot motion intent with augmented reality," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2018, pp. 316–324.

[19] T. Chakraborti, S. Sreedharan, A. Kulkarni, and S. Kambhampati, "Projection-aware task planning and execution for human-in-the-loop operation of robots in a mixed-reality workspace," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4476–4482.

[20] G. Bolano, C. Juelg, A. Roennau, and R. Dillmann, "Transparent robot behavior using augmented reality in close human-robot interaction," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–7.

[21] D. Krupke, F. Steinicke, P. Lubos, Y. Jonetzko, M. Görner, and J. Zhang, "Comparison of multimodal heading and pointing gestures for co-located mixed reality human-robot interaction," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.

[22] R. S. Sutton, D. Precup, and S. Singh, "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning," *Artificial intelligence*, vol. 112, no. 1-2, pp. 181–211, 1999.

[23] C. Diuk, A. Cohen, and M. L. Littman, "An object-oriented representation for efficient reinforcement learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 240–247.

[24] U. 3D. Unity Asset Store. [Online]. Available: https://assetstore.unity.com/

[25] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013.

[26] D. Whitney, E. Rosen, D. Ullman, E. Phillips, and S. Tellex, "ROS reality: A virtual reality framework using consumer-grade hardware for ROS-enabled robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.

[27] M. C. Medlock, D. Wixon, M. Terrano, R. Romero, and B. Fulton, "Using the rite method to improve products: A definition and a case study," *Usability Professionals Association*, vol. 51, 2002.

[28] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek, "The trimmed iterative closest point algorithm," in *Object recognition supported by user interaction for service robots*, vol. 3. IEEE, 2002, pp. 545–548.