

PPS-Tags: Physical, Perceptual and Semantic Tags for Autonomous Mobile Manipulation

Hai Nguyen¹, Travis Deyle¹, Matt Reynolds², and Charles C. Kemp¹

¹Healthcare Robotics Lab, Georgia Institute of Technology, USA

²Department of Electrical and Computer Engineering, Duke University, USA

Abstract—For many promising application areas, autonomous mobile manipulators do not yet exhibit sufficiently robust performance. We propose the use of tags applied to task-relevant locations in human environments in order to help autonomous mobile manipulators physically interact with the location, perceive the location, and understand the location’s semantics. We call these tags physical, perceptual and semantic tags (PPS-tags). We present three examples of PPS-tags, each of which combines compliant and colorful material with a UHF RFID tag. The RFID tag provides a unique identifier that indexes into a semantic database that holds information such as the following: what actions can be performed at the location, how can these actions be performed, and what state changes should be observed upon task success?

We also present performance results for our robot operating on a PPS-tagged light switch, rocker light switch, lamp, drawer, and trash can. We tested the robot performing the available actions from 4 distinct locations with each of these 5 tagged devices. For the light switch, rocker light switch, lamp, and trash can, the robot succeeded in all trials (24/24). The robot failed to open the drawer when starting from an oblique angle, and thus succeeded in 6 out of 8 trials. We also tested the ability of the robot to detect failure in unusual circumstances, such as the lamp being unplugged and the drawer being stuck.

I. INTRODUCTION

Autonomous mobile manipulation within human environments represents both an exciting opportunity for new robotic applications and a grand challenge for robotics [11]. Although researchers continue to make progress in this area, autonomous mobile manipulators do not yet exhibit sufficiently robust performance to support many promising applications. For example, if assistive mobile manipulators could robustly operate within real homes for extended periods of time, they could provide valuable in-home assistance. We see the critical deficiencies of current robots as falling into the following three inter-related categories:

Physical:

The robot’s mechanical structure may be poorly matched to the task. For example, a robot with a primitive gripper may be unable to pull on a recessed handle, or a small mechanism may be too difficult to grasp reliably.

Perceptual:

The robot may be unable to reliably perceive the task-relevant features required for consistent success at the task. For example, a thin pull chain may be too small for a robot’s laser range finder to

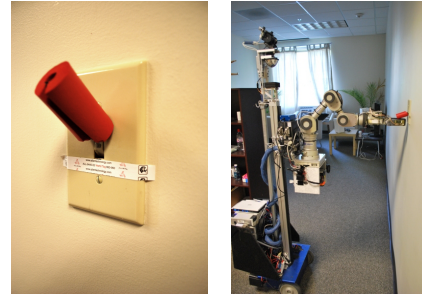


Fig. 1. *Left*: Example of a PPS-tag for a light switch *Right*: Robot using the tag.

detect, or the robot may be unable to reliably detect drawer handles due to the wide variety of handles found in human environments.

Semantic:

The robot may be unable to infer the task-relevant semantics, such as what actions it can perform with a particular mechanism or the implications of those actions. For example, the robot may not realize that it can pull on a chain to operate a lamp and that this should either increase or decrease the light from the lamp.

Many approaches seek to address one or more of these shortcomings. In this paper, we propose augmenting environments to directly help robots with these three challenges. Specifically, we present PPS-tags, which stands for physical, perceptual, and semantic tags. We have designed these tags to be affixed to sparse task-relevant locations in the environment in order to help the robot physically interact with the location, perceive the location, and understand the location’s semantics.

While we ultimately hope to develop robots that will not require modifications of the environment, we believe PPS-tags offer several advantages at this time. For example, PPS-tags have the potential to accelerate the deployment of autonomous mobile manipulators in real-world applications. This could have societal and economic benefits. It could also benefit robotics research by providing data from real-world usage scenarios. Also, PPS-tags could represent a beneficial path for system development and research. One can imagine first developing a robotic system that uses PPS-tags and then



Fig. 2. *Top*: PPS-tags affixed to (from left to right): flip-type light switch, ADA-compliant rocker-type light switch, lamp pull chain, cabinet drawer, and trashcan. *Bottom*: EL-E manipulating the corresponding PPS-tags.

gradually removing them or altering them in conjunction with the development of improved mechanical, perceptual, or semantic capabilities. Similarly, researchers can use PPS-tags to immediately explore system-level questions, rather than waiting for the solutions to long standing problems such as object recognition. Additionally, we believe PPS-tags might enable simple inexpensive robots to perform complex tasks.

II. RELATED WORK

Here we provide a brief overview of related work.

A. Robots in Augmented Environments

People often alter environments for robots. For example, in factories people create robotic work cells matched to the tasks performed by the robot. There are also many examples of environmental modification for robots outside of industrial settings. For example, most high performing systems in RoboCup competitions depend on environments that are easy to perceive with color vision [2], [21]. Also, many robots have depended on perceptual augmentation of the environment, such as with ARTags and QR tags [8], [9]. Roomba owners roombarize their homes, a process that often involve changing furniture layouts, cleaning up wires, and tucking in rug tassel [20]. People sometimes attach fabric to the handles of doors and drawers so that service dogs can operate them. We have previously demonstrated the use of towels as a physical and perceptual aid for a robot [16]. In contrast to prior work, PPS-tags combine physical, perceptual, and semantic assistance to enable a robot to perform a variety of tasks using similar behaviors. Unlike methods that depend on complex sensing, such as cameras throughout the environment [22], PPS-tags can be simple,

sparsely distributed, inexpensive, and independent from one another. In our current implementation, the robot does not require detailed models of the environment nor the tagged objects, and instead uses sparse task-relevant information.

B. RFID-assisted Robots

Radio-frequency identification (RFID) represents another important example of environmental augmentation for robots. Due to the low cost of tags and the opportunity for non line-of-sight perception, RFID tags have enjoyed a great amount of attention in robotics. Researchers have developed robots that can navigate to RFID-tags, localize them, and build maps with respect to them [14], [5], [13]. Researchers have also explored opportunities for associating semantic information with the unique identifier provided by an RFID-tag. Using XML profiles, the authors of [12] defined object properties such as weight, and grip force for a table mounted robot. Ha *et al* [4] proposed a knowledge architecture based on the semantic web language, OWL-S, to describe objects, possible actions, and the expected effects of actions. Baeg [1] described a smart home environment with interoperating devices such as RFID enabled tables, shelves, and mobile robots. Jang, Sohn, and Cho [6] presented an architecture for associating semantic labels and properties such as indicating what areas are restricted within a physical space. Hidayat *et al.* [19] proposed that objects should be tagged with their affordances.

Although many researchers have previously suggested that RFID-indexed databases could be used by robots, there is a lack of published results describing real robots making use of the proposed information. The authors of [15] may be the first to have implemented their architecture on a

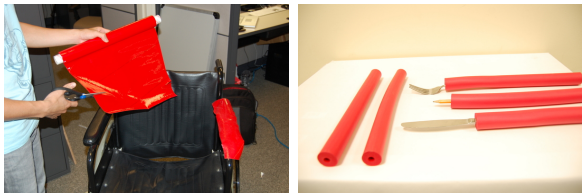


Fig. 3. Two commonly sold assistive devices used in this research. *Left:* High-friction Dycem polymer. *Right:* Compliant, slip resistant foam tubing.

mobile manipulator, and is the only work we have found that describes a real mobile manipulator making use of RFID-indexed semantic information. Their robot moved a cup and a chair using object properties loaded from an RFID indexed database.

In contrast to previous work, we have implemented our system with a mobile manipulator and tested it with 5 different devices. We have found that a relatively simple semantic structure with only a handful of entries is sufficient to support these tasks. We pursued a bottom-up design strategy in which the goal of the robot performing specific, well-defined tasks dictated the contents of the semantic database. PPS-tags also combine this semantic assistance with physical, and perceptual help.

III. PPS-TAGS ILLUSTRATED IN THREE EXAMPLES

In this section, we present our concept for PPS-tags and our current implementation. The PPS-tag concept is general and could take many forms. A PPS-tag is a tag that can help a robot physically, perceptually, and semantically.

To operate within a PPS-tagged environment, a mobile manipulator needs to be able to (1) physically interact with the tags, (2) perceive the tags, and (3) detect the tags' identities so that corresponding queries can be made in an associated semantic database. Ideally, a robot would only need to perceive, understand, and physically interact with these tags in order to perform useful tasks.

A. Physical: Manipulating High-Friction, Well-Sized, Compliant Materials

Each of the three tags shown in Figure 2 (first, second, and fourth panel) provides a different form of physical assistance. All of them are compliant with high friction, and, except for the red patch, significantly increase the target volume over which the robot can successfully grasp.

For the first type of PPS-tag, we use a non-slip, compliant red foam tube. This tubing is normally used to make eating utensils and cylindrical objects, such as toothbrushes and pencils, easier for people with motor-impairments to manipulate. We purchased this foam tube from an online store (Rehabmart.com) that supplies materials and devices to assist people with physical disabilities.

For the second PPS-tag example, we use patches of red Dycem polymer, a high friction, compliant material commonly affixed to wheel chairs and walkers to prevent grip slippage (Figure 3). We purchased this material from the same store.

For the third PPS-tag, we use a red towel. As in our previous work, the towel is affixed to drawers and doors [16]. In contrast to our previous paper, which included results for two drawers and two doors, we only present results from one drawer in this work. The towel provides a large compliant target for grasping, which is easier for our robot to grasp onto than the large variety of possible handles found in human environments.

B. Perceptual: Segmenting Point Clouds Based on Registered Color Images

For the work in this paper, we assume that our robot is first given a rough 3D location in the environment near which it is supposed to perform an action. This 3D location could come from a number of sources, such as long-range RFID localization [3]. In this work, we provide the 3D location using our laser pointer interface, where the user selects 3D locations by briefly illuminating them with a laser pointer [10].

All three types of tags share a very similar red color that occupies a large area. Our robot uses the same method to perceive all three tag types. First, using a camera and a tilting laser range finder that are calibrated and registered with one another, our algorithm acquires a color image and a 3D point cloud. The image is taken using the camera's flash. The next step segments parts of the 3D point cloud that correspond with red in the camera image.

The segmentation proceeds in the following order. First, our algorithm performs a 2D segmentation of red patches in the image using minimum and maximum thresholds defined in HSV space, as describe in [7]. The next step post processes this raw segmented image with a series of morphological operations: hole filling, closing, then opening. Using the known 3D rigid body transformation between the point cloud and the camera, the algorithms projects the 3D point cloud into the color segmented image. Points projected onto red-segmented regions are kept, all others are discarded. From these 3D points with associated red labelings, the algorithm constructs a 3D occupancy grid (resolution 1 cm^3). It uses this grid to separate the point cloud into 3D connected components keeping only the centroid of the connected component closest to the target 3D location.

The large red materials make this simple perceptual algorithm effective even at a distance.

C. Semantic: An RFID-Indexed Database with Grounded Semantics

In addition to the red compliant material, each of our three examples of PPS-tags includes a self-adhesive UHF RFID tag. Our robot reads this tag using short-range RFID antennas embedded in its fingers. Using this unique identifier, the robot queries a semantic database. Currently, this database is implemented as a series of nested hashtables with the first level of hashtables indexed by the RFID tag's unique identifier. A sample of one top-level entry in the database for a rocker switch can be seen in Figure 4.

```

{
  'properties': {'type': 'ada light switch',
                'name': 'A D A light switch 1',
                'pps_tag': 'dycem',
                'change': 'overall brightness',
                'switch_travel': 0.02,
                'height': 1.22,
                'on_plane': True,
                'direction': 'up',
                'ele': {'color_segmentation':
                      [[34, 255], [157, 255],
                       [0, 11]]},
                },
  'actions': {'off': 'push_bottom',
              'on': 'push_top'},
  'push_bottom': {'force_threshold': 3.0,
                  'height_offset': -0.02,
                  'ele': {'gripper': 5}},
  'push_top': {'force_threshold': 3.0,
               'height_offset': 0.02,
               'ele': {'gripper': 5}}
}

```

Fig. 4. A semantic database entry for operating an ADA light switch. The current database is written in Python.

We focus on grounded semantics for robot manipulation. By this, we mean that we restrict the semantic database to hold information that directly informs the robot’s manipulation behaviors. Each object-specific entry in our database contains three main components: properties, actions, and details of how to perform each action. In our work, we have designated “properties” as a place for information about the object that is not specific to an action (Figure 4). Within “actions”, we map user-friendly names for the actions that can be performed to associated robot behaviors. Finally, in separate hashtables (e.g. “push_bottom”, “push_top”) we store parameters used by each of the robot’s behaviors.

To illustrate the semantic database and its use by the robot, we now describe the example entry in detail, see Figure 4.

1) *properties*: In “type”, we store the class of object, such as “ada light switch”, “light switch”, or “trash can”. In “name”, we store a unique name that is specific to this particular object instance, such as “A D A light switch 1”, “light switch 1”, or “trash can 1”. Both of these levels of naming, class and instance, could potentially be useful to the robot, such as when collecting data from experience which may relate to the specific instance or the class of object being used. “pps_tag” defines the type of tag being used, such as “dycem”, “towel”, or “foam tube”. “change” describes the change in state that should be observed upon using the object successfully, such as “overall.brightness” for lighting and “location” for the drawer. “direction” tells the robot where to look to observe this state change, such as “up” for the light switches and “forward” for the lamp. “ele” contains a hash table with information specific to the robot EL-E. In this case it holds the HSV color segmentation boundaries that segment the PPS-tag with EL-E’s camera.

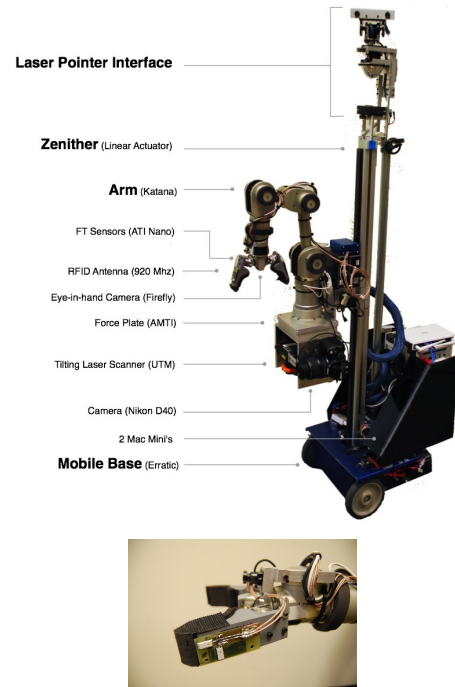


Fig. 6. *Top*: Our robot EL-E (pronounced “Ellie”). *Bottom*: Fingers with short range RFID antennas.

2) *actions and behaviors*: For this “ada light switch”, the two associated actions are turning the light “on” and turning the light “off”. These map to the “push_top” and “push_bottom” behaviors respectively. Each of these behaviors also has an entry which stores information important to performing the action. For example, “push_bottom” holds information critical to pushing the bottom of the rocker switch in order to turn the light off. It has the entries “height_offset” with a value of -0.02 meters, and “force_threshold” with a value of 3 Newtons. These describe how far below the center of the PPS-tag to push and the force to apply when pushing. The “ele” entry for the “push_bottom” behavior how close the opening angle that should be used by EL-E’s gripper when performing this action. 5 degrees places EL-E’s gripper in a pinching configuration that is useful for pushing the button.

IV. A ROBOT AND BEHAVIORS THAT USE THE TAGS

Within this section, we describe the robot we used to evaluate our implementation of PPS-tags and its behaviors.

A. Our Platform

We performed this work on the robot EL-E as shown in Figure 6 and described in previous papers, such as [16]. The most pertinent sensors to this work are EL-E’s laser pointer interface, DSLR camera (Nikon D40), tilting laser range finder (Hokuyo UTM-30LX mounted on a Robotis Dynamixel RX-28 servo motor), finger-mounted force-torque sensors, palm-mounted IR range sensor, and finger-mounted RFID readers. The laser pointer interface detects a green laser spot from a laser pointer held by the user and estimates its 3D location [10].

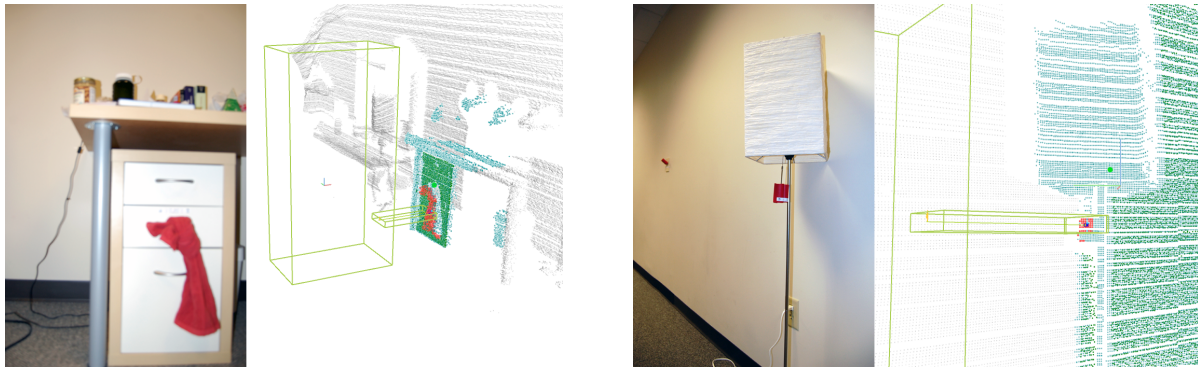


Fig. 5. *Left Pair:* Camera image and point cloud used to segment the drawer’s PPS-tag and find an approach direction. *Right Pair:* Camera image and point cloud used to segment the lamp’s PPS-tag and find an approach direction. (Key for point cloud colors: Cyan points lie within the volume of interest. Red points were found to correspond to the PPS-tag. Dark green points denote the plane found closest to the PPS-tag’s centroid. The yellow-green bounding boxes check for potential collisions with EL-E’s body and arm.)

In order to associate a unique identifier with each tag, we equipped EL-E’s fingers with RFID sensing capabilities. On each tagged object we use an Alien Technologies’ Gen2 “Squiggle” Ultra High Frequency (UHF) tag. These tags are passive, economical (20 cents each), easily applied, and able to communicate with our short range antennas. On each of EL-E’s fingers we used a pair of ceramic 920MHz microstrip antennas (Johnson Technology part number 0920AT50A080E) mounted at 90° offsets to provide comprehensive RF coverage for each finger and expected read ranges of up to 15 cm. However, in practice we have observed read ranges of up to 50 cm. The robot uses the tag with the strongest receive signal strength indicator (RSSI) in order to avoid ambiguities that might be caused by this read range. Since metals are opaque to RF signals, we fabricated fingers for EL-E out of 3D-printed ABS plastic. Although we do not use the capability in this paper, the same RFID tags support long-range reading and localization using long range antennas [3]. These capabilities would complement the work in this paper, allowing us to localize, approach, and read semantic information from a tag that is across a room.

B. Behaviors

We now describe EL-E’s behavior after the user selects a 3D location in the world using the laser pointer interface. We have broken the task of approaching this 3D location and manipulating it into four phases. In the first phase, when EL-E is further than a threshold distance away (1.0 m), the robot will drive towards the given location. If it travels for longer than a threshold distance, it asks for the user to designate the location again in order to reduce 3D estimation errors. In the second phase, described in more detail in section IV-B.1.a, EL-E uses its laser range finder and camera to locate and drive to the selected PPS-tag. It orients itself so that it is perpendicular to the estimated flat surface on which the selected PPS-tag rests. In the third phase, EL-E drives slowly forward with its gripper closed and fingers pointed forward. It stops when its fingers are estimated to be within 10 cm of the selected location (detailed in section IV-B.1.b), or the force on the fingers goes above a threshold. Once EL-E stops,

it attempts to read the unique identifier for the nearest RFID tag by reading the ID of the tag with the strongest signal strength.

In the final phase, EL-E reads the entry in the semantic database associated with the PPS-tag’s RFID. If there is only one possible action, EL-E performs it. If there are two, EL-E asks the user to select between the two choices by pointing the laser pointer either up or down, such that the laser point is either more than 50cm off of the ground or less. Depending on the object identified, users can select between on, off (section IV-B.2.b), pull-back (section IV-B.2.c), push (section IV-B.2.d), pull-lamp (section IV-B.2.e), or drop (section IV-B.2.f). After selection, the robot executes the selected behavior, asks the user whether if it was successful in its execution, and then records this response in the database along with the time when the task started, the time that it finished, the force-torque information measured during execution, and additional behavior-specific information.

1) *Navigation to the PPS-tag:* In the following two sections we describe in detail the process that EL-E uses to navigate to the user-selected object in preparation for manipulating it.

a) *Approaching the PPS-tag:* As many task-relevant locations in human environments are situated on or near vertical surfaces (e.g., walls and the front sides of appliances, doors, and drawers). We heuristically assume that an effective way to approach the selected PPS-tag will be the direction perpendicular to the closest vertical surface.

After orienting toward and approaching the 3D location selected by the user, the robot uses its tilting laser range finder to acquire a point cloud. The robot then defines a cylindrical volume of interest (VOI) around the 3D location selected by the user, such that the axis of the cylinder is parallel to gravity and passes through the selected location. Within this VOI, the robot uses MLSAC, a variant of RANSAC which attempts to find all planes in the portion of the point cloud that falls within the VOI. For MLSAC we use the implementation provided through the ROS personal-robots repository [18], [17]. The robot then throws out all of the planes it found with fewer than 100 points and selects

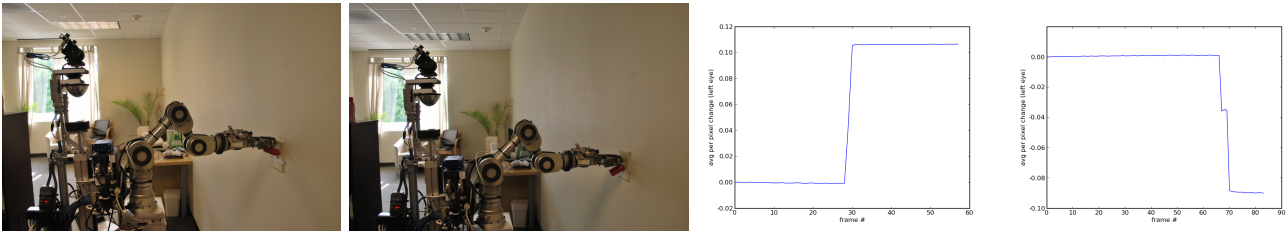


Fig. 7. *Left Pair*: EL-E turning off a light switch. *Right Pair*: Brightness changes resulting from the light being switched off and from the light being switched on.

the remaining plane whose member points come closest to the user-selected 3D location.

Next, the robot finds the location of the PPS-tag closest to the user-select location using the algorithm previously described in section III-B. Given the estimated location and orientation of the plane and the estimated location of the PPS-tag, the robot calculates a waypoint 50 cm from the selected tag in the direction perpendicular to the plane. The robot then checks if it can be centered at the waypoint facing the PPS-tag with its arm extended without colliding with points in the point cloud. If this test passes, EL-E drives towards the waypoint. Once it reaches the waypoint, it orients to the PPS-tag.

b) Determining the PPS-tag's Identity: While driving towards the waypoint, EL-E updates its estimates of the PPS-tag's location using odometry. This results in substantial accumulated error, so the robot now performs an additional navigation step so that it can read the PPS-tag's RFID and manipulate the PPS-tag. To achieve this, EL-E visually servos to the PPS-tag using its eye-in-hand camera. While readjusting its pose, EL-E monitors the forces on its fingers, and stops if it detects a collision. When EL-E's arm is estimated to be 10 cm from the PPS-tag, it closes its gripper to face the finger mounted antennas forward and attempts to read the RFID tag.

2) Manipulating the PPS-tag: We now describe the behaviors used by EL-E to manipulate PPS-tags.

a) Close Range Alignment: After the two navigation steps described above, the robot assumes that its end effector is approximately 10 cm away from the PPS-tag. In order to refine its pose, EL-E moves forward until it detects contact with the force/torque sensors in the base of its fingers or near contact with the IR range sensor in its palm. Upon detection, EL-E backs off a fixed distance (8 to 12 cm depending on the behavior). In our tests, contact or near contact was typically made with the tag or the vertical plane behind the tag.

b) Light Switch: To operate a light switch, EL-E moves the carriage up (or down depending on the command), closes the gripper (but not all the way), moves the gripper forward until contact has been made with the wall (using a 2 N threshold), moves the gripper away from the wall by 2 cm (to clear the plate on which the light switch is mounted), then moves the carriage down (or up) using torque control with the gripper extended, and stops when the maximum force on EL-E's gripper is greater than 12 N or has traveled 15 cm.

To monitor the effects of using the light switch, EL-E takes

a picture of the expected location of the light source with its stereo camera prior to, and after moving the carriage up or down. To determine the effect of its attempt to use the light switch, EL-E takes the average intensity of the image before the action and subtracts the average intensity of the image after the action. If the magnitude of the change is greater than a threshold and the result is positive, EL-E concludes that the light has been turned off, and if it is negative EL-E concludes that the light has been turned on.

c) Pull Back: To pull on a towel, EL-E first moves its arm forward to grasp the towel. As there is only one degree of freedom for both of EL-E's fingers, the towel grasping behavior in our previous work [16] would sometimes cause large forces to accumulate if EL-E's grippers were not positioned close to the center of the towel. To some extent, this issue was mitigated by the compliance provided by the towel. In this work, we have implemented a new grasping behavior that uses the force-torque sensors to move the end effector laterally while grasping to correct for small misalignments. This grasping behavior has the effect of centering the towel in the middle of EL-E's gripper making it more likely that forces on EL-E's fingers will be distributed evenly across the two fingers as the towel is pulled backwards.

If the grasping behavior detects that it has been successful (forces on both of EL-E's fingers exceed 2 N), then EL-E proceeds to pull on the towel by moving backwards with its mobile base. EL-E moves back in steps of 20 cm, stopping when either a force threshold is exceeded (drawer is fully opened), the force on the fingers drops below a threshold (fingers lose their grip), or the robot has moved back farther than a defined distance (drawer is fully open). At the end of each complete pulling step, EL-E runs the towel grasping behavior again to maintain its grip on the towel.

During this pull back behavior, EL-E records the displacement of its end effector between the time when it successfully grasps the towel and when the robot either loses its grip or finishes pulling. If this distance is greater than 50% of the expected pull distance, EL-E declares its action successful. Otherwise it declares a failure. The robot records this displacement in its semantic database for future use.

d) Push: The goal of the push behavior is for the robot to apply a force normal to the detected plane upon which the PPS-tag rests. The push behavior can be used to either activate a light switch or close a drawer. EL-E first sets its gripper to the settings specified in the semantic database. In this work, for the ADA light switch, EL-E makes

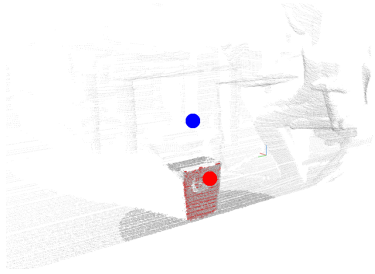


Fig. 8. Visualization of the drop behavior. The segmented planar front face of the trash can is in red. The large red dot is the centroid of the detected PPS-tag, and the large blue dot is the location at which EL-E will attempt to release the object.

the gripper more appropriate for poking by closing it. For pushing drawers, EL-E’s gripper fully opens to maximize the chances that its end-effector will make contact with the drawer’s surface. Next, EL-E moves its end-effector forward stopping if a force greater than 2 N is detected or the IR sensor in the palm reports an obstruction. Having made contact, or coming close to it, EL-E then pushes forward with its base for the requested distance or until a force threshold is exceeded.

Prior to pushing, EL-E loads the appropriate success state detector for this PPS-tag from the semantic database. At this time, the two possible success criteria are detecting brightness changes and detecting how far forward EL-E has pushed, for the light switch and drawer respectively. As in the other behaviors, the push behavior records whether it has succeeded and the information used to make this decision in the semantic database.

e) Pull Lamp: In this behavior EL-E pulls on the pull chain of a commonly available IKEA free standing living room lamp. For this behavior, EL-E has only moved back 8 cm after contact to place the PPS-tag in the forward part of its gripper. EL-E then closes its gripper, stopping when a threshold force has been reached. After gripping the pull chain, EL-E uses the same carriage control as in the light switch operation to apply a downward force directly on the lamp’s chain, stopping when EL-E’s fingers either detect a force greater than 10 N or have moved down more than 7 cm. As with the other lighting related behaviors, the pull lamp behavior monitors the change in lighting to determine success or failure. Unlike with the switches that operate ceiling lights, the lamp pull chain points the stereo camera forward prior to performing the action. The final step is to record the result of this behavior along with the captured images.

f) Drop: The goal of the drop behavior is for EL-E to take an object from its hand and drop that object into a user indicated container. The object in EL-E’s hand often obstructs the eye-in-hand camera, so visual servoing is not used to refine the robot’s pose prior to executing this behavior (section IV-B.2.a). Instead, EL-E uses its laser range finder to calculate a location above the container where the object can be dropped.

In a manner similar to the initial localization of the PPS-

tag, EL-E first uses its tilting laser range finder to scan the container, segment out all planes in a cylindrical volume around the PPS-tag, select the closest plane to the PPS-tag, calculate the orientation of the plane, and find a point above the container from which to drop the object.

After this, EL-E drives towards the container stopping 45 cm from the drop location, moves its end effector to the location, and releases the object. For this behavior, EL-E detects success based on whether or not it senses the object in its grasp using its finger-mounted force-torque sensors and the palm-mounted IR range sensor.

V. EXPERIMENTAL EVALUATION

We now present results from our tests of EL-E’s effectiveness in operating PPS-tagged devices. Our first goal was to test the behaviors multiple times to estimate their reliability. Our second goal was to evaluate the system’s dependence on the relative orientation of the robot to the object being operated. Thirdly, we wanted to test EL-E’s ability to recognize when it failed to operate a device. All the trials that we report here were performed in the Healthcare Robotics Lab using standard office fluorescent lighting.

In the first set of trials, we varied the tagged device used by the robot, the robot’s position with respect to the device, and the action selected for a total of 32 trials. At the beginning of each trial, we positioned EL-E 1.5 meters away from the device’s PPS-tag in one of four directions. The robot was always facing towards the tag at the beginning of the trial. We then provided EL-E with a 3D location via the laser pointer interface. If multiple actions were available, we would also select the action to perform using the laser pointer interface.

In detail, for the regular and ADA light switch (rocker) we placed EL-E on evenly spaced locations along the 1.5 meter radius half-circle centered at the light switch. We placed the lamp, drawer, and trashcan next to a wall and performed the same procedure. However, we also placed the lamp such that its pull chain faced outwards in the direction perpendicular to the wall, as required by our current implementation.

In the second set of trials, we tested EL-E on an unplugged lamp, a stuck drawer, and a sticky object to test the robot’s ability to detect failure. The sticky object used in this case was a sphere of double-sided tape in order to simulate potential failures in releasing normal objects. In this case, we defined success as EL-E attempting to perform the task, performing what would usually be a successful action, and reporting that it was not able to perform the task as indicated.

A. Results

We present the results from these two sets of trials in table I. In the first set, EL-E was able to carry out all tasks with the exception of a pulling and a pushing trial on the drawer.

In contrast to our previous work where EL-E had an 80%-90% success rate operating the drawers, our current implementation failed at pushing and pulling once for each of the trials in the first set of experiments. These errors corresponded with oblique angles of the robot relative to the front face of the drawer. We believe this led to failures

TABLE I
EXPERIMENTAL RESULTS

PPS-Tagged Object	Scenario	Success Rates
Flip-type light switch	Switching on	4/4 = 100%
	Switching off	4/4 = 100%
ADA-Compliant rocker-type light switch	Switching on	4/4 = 100%
	Switching off	4/4 = 100%
Lamp	Switching on	4/4 = 100%
	Detect switching failure	4/4 = 100%
Trash can	Drop object into	4/4 = 100%
	Detect dropping failure	4/4 = 100%
Cabinet Drawer	Pull open	3/4 = 75%
	Push closed	3/4 = 75%
	Detect pulling failure	2/4 = 50%
	Detect pushing failure	3/4 = 75%

due to the plane on the side of the drawer being selected as the relevant backing plane, rather than the front plane. This also led to failures on the error detection tests. We plan to correct this error in future work.

VI. DISCUSSION AND FUTURE DIRECTIONS

Our work makes two main contributions. First, we have presented the concept of PPS-tags which provide physical, perceptual, and semantic help to robots. Second, we presented three examples of PPS-tags along with a set of robotic behaviors that enabled us to evaluate their performance on a set of tasks.

We have only presented three PPS-tag types. It is not hard to imagine tags that have different and potentially better properties. For example, tags with more compact and aesthetic designs would be beneficial. Also, it could be useful for the tags to provide a 6D frame of reference, like ARTags, rather than only a position with an implied orientation coming from a nearby plane. In addition, with tasks that involve multiple task-relevant locations, such as carrying a two-handled tray or performing tasks in the kitchen, we expect that richer semantic information that enables tags to reference one another could be valuable.

With the semantic database, we now have the ability to gather information about the robot's interaction with each device over time, which could potentially serve as a resource for self-guided learning. More generally, multi-robot coordination through the tags could be valuable, such as forms of stigmergy. Having grounded, hierarchical semantic information that can be tailored to individual robots or abstracted to groups of robots might also prove valuable. One could imagine a smarter, more sensor rich robot travelling through the environment, tagging locations, and recording relevant information for use by less sophisticated robots. More generally, we expect that exploring the potential for simple robots to operate PPS-tagged devices would be worthwhile.

VII. ACKNOWLEDGMENTS

The authors thank the ROS community for its software and prompt assistance. We also gratefully acknowledge support

from Willow Garage, NSF grant CBET-0932592, and NSF grant IIS-0705130.

REFERENCES

- [1] S.-H. Baeg, J.-H. Park, J. Koh, K.-W. Park, and M.-H. Baeg. Robo-maidhome: A sensor network-based smart home environment for service robots. In *16th IEEE International Conference on Robot and Human Interactive Communication*, 2007.
- [2] J. Bruce, T. Balch, and M. Veloso. Fast and Inexpensive Color Image Segmentation for Interactive Robots. In *International Conference on Intelligent Robots and Systems (IROS)*, 2000.
- [3] T. Deyle, H. Nguyen, M. Reynolds, and C. Kemp. Rf vision: Rfid receive signal strength indicator (rssi) images for sensor fusion and mobile manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.
- [4] Y.-G. Ha, J.-C. Sohn, and Y.-J. Cho. Service-oriented integration of networked robots with ubiquitous sensors and devices using the semantic web services technology. In *International Conference on Intelligent Robots and Systems (IROS)*, 2005.
- [5] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. Mapping and localization with rfid technology. In *ICRA*, 2004.
- [6] M. Jang, J.-C. Sohn, and Y. Cho. Building semantic robot space based on the semantic web. In *IEEE International Conference on Robot and Human Interactive Communication*, 2007.
- [7] P. Kakumanua, S. Makrogiannisa, and N. Bourbakis. A survey of skin-color modeling and detection methods. In *Pattern Recognition*, 2007.
- [8] H. Kato, M. Billingham, K. I. I. Poupyrev, and K. Tachibana. Virtual object manipulation on a table-top ar environment. In *International Symposium on Augmented Reality*, 2000.
- [9] R. Katsuki, J. Ota, Y. Tamura, T. Mizuta, T. Kito, T. Arai, T. Ueyama, and T. Nishiyama. Handling of object with marks by a robot. In *International Conference on Intelligent Robots and Systems*, 2003.
- [10] C. C. Kemp, C. D. Anderson, H. Nguyen, A. J. Trevor, and Z. Xu. A point-and-click interface for the real world: Laser designation of objects for mobile manipulation. In *International Conference on Human-Robot Interaction*, 2008.
- [11] C. C. Kemp, A. Edsinger, and E. Torres-Jara. Challenges for robot manipulation in human environments. *IEEE Robotics & Automation Magazine*, 14(1):20–29, March 2007.
- [12] B. K. Kim, M. Miyazaki, K. Ohba, S. Hirai, and K. Tanie. Web services based robot control platform for ubiquitous functions. In *International Conference on Robotics and Automation (ICRA)*, 2005.
- [13] A. Kleiner, J. Prediger, and B. Nebel. Rfid technology-based exploration and slam for search and rescue. In *IROS*, 2006.
- [14] V. Kulyukin, C. Gharpure, and J. Nicholson. Robocart: Toward robot-assisted navigation of grocery stores by the visually impaired. In *International Conference on Intelligent Robots and Systems (IROS)*, 2005.
- [15] A. Ming, Z. Xie, T. Yoshida, M. Yamashiro, C. Tang, and M. Shimojo. Home service by a mobile manipulator system - system configuration and basic experiments. In *International Conference on Information and Automation*, 2008.
- [16] H. Nguyen and C. C. Kemp. Bio-inspired Assistive Robotics: Service Dogs as a Model for Human-Robot Interaction and Mobile Manipulation". In *IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BIROB)*, 2008.
- [17] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, R. W. Eric Berger, and A. Ng. ROS: an open-source Robot Operating System. In *Open-Source Software workshop of the International Conference on Robotics and Automation (ICRA)*, 2009.
- [18] R. B. Rusu, W. Meeussen, S. Chitta, and M. Beetz. Laser-based perception for door and handle identification. In *Proceedings of International Conference on Advanced Robotics*, 2009.
- [19] B. K. Sidiq S. Hidayat and K. Ohba. Affordance-based ontology design for ubiquitous robots. In *IEEE International Symposium on Robot and Human Interactive Communication*, 2008.
- [20] J.-Y. Sung, R. Grinter, H. Christensen, and L. Guo. Housewives or technophiles?: Understanding domestic robot owners. In *Proceedings of the ACM Conference on Human Robot Interaction*, 2008.
- [21] Z. Wasik and A. Saffiotti. Robust color segmentation for the RoboCup domain. In *International Conference on Pattern Recognition*, 2002.
- [22] A. Williams, D. Xie, S. Ou, R. Grupen, A. Hanson, and E. Riseman. Distributed smart cameras for aging in place, 2006.