

# A Generalized Framework for Opening Doors and Drawers in Kitchen Environments

Thomas Rühr<sup>1</sup>, Jürgen Sturm<sup>2</sup>, Dejan Pangercic<sup>1</sup>, Michael Beetz<sup>1</sup>, Daniel Cremers<sup>2</sup>

**Abstract**—In this paper, we present a generalized framework for robustly operating previously unknown cabinets in kitchen environments. Our framework consists of the following four components: (1) a module for detecting both Lambertian and non-Lambertian (i.e. specular) handles, (2) a module for opening and closing novel cabinets using impedance control and for learning their kinematic models, (3) a module for storing and retrieving information about these objects in the map, and (4) a module for reliably operating cabinets of which the kinematic model is known. The presented work is the result of a collaboration of three PR2 beta sites. We rigorously evaluated our approach on 29 cabinets in five real kitchens located at our institutions. These kitchens contained 13 drawers, 12 doors, 2 refrigerators and 2 dishwashers. We evaluated the overall performance of detecting the handle of a novel cabinet, operating it and storing its model in a semantic map. We found that our approach was successful in 51.9% of all 104 trials. With this work, we contribute a well-tested building block of open-source software for future robotic service applications.

## I. INTRODUCTION

Mobile manipulation robots often need to interact with articulated objects to accomplish their service tasks. For example, a household robot that is given the task to prepare a meal has to be able to open the refrigerators and drawers in the kitchen. In recent years, much progress has been made in this domain [1]–[4]. However, the evaluation of these systems was in most cases limited to a single robot operating in a single kitchen with only small variations in the appearance of the handles, the illumination conditions and kinematic properties of cabinets. Until recently, the cross-validation of such systems by others was difficult as the robot hardware was often proprietary and highly specialized. Therefore, software developed in one research lab was difficult to be evaluated on a different robot in another lab. This situation has changed fundamentally: novel frameworks like ROS have fostered software re-use between labs and robots significantly, and standardized mobile manipulation platforms like the PR2 have further accelerated this process.

This paper is the result of a collaboration of three PR2 beta sites (ALU Freiburg, BOSCH and TUM) with the goal to create a reliable and well-tested software framework that enables mobile manipulation robot to autonomously operate

This work was partially funded by the cluster of excellence CoTeSys.

<sup>1</sup>T. Rühr, D. Pangercic and M. Beetz are with the Intelligent Autonomous Systems group, Computer Science Department, Technische Universität München {ruehr, pangercic, beetz}@cs.tum.edu

<sup>2</sup>J. Sturm and D. Cremers are with the Computer Vision and Pattern Recognition group, Computer Science Department, Technische Universität München {sturmju, cremers}@cs.tum.edu

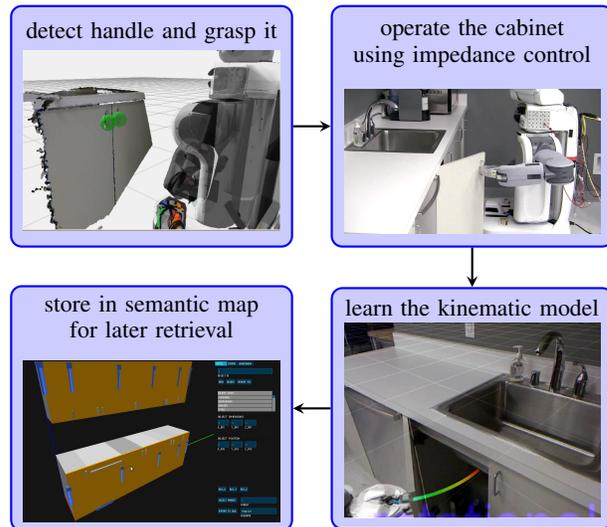


Fig. 1. We provide a general framework for operating cabinets in kitchen environments that we evaluated in five different kitchens across three different institutions. The overall success rate of the PR2 robot averaged over all kitchens is 51.9%.

cabinets. We have combined our expertise in 3D perception and semantic mapping [5] and real-time control for operation of articulated objects [6], [7] to create an integrated system for operating cabinets. Our system enables a robot

- 1) to detect, localize and grasp the handles of kitchen cabinets,
- 2) to operate cabinets using impedance control while learning their kinematic models and appearance,
- 3) to integrate the learned model into the robot’s semantic map of the environment, and
- 4) to re-use this kinematic model for future encounters.

The key contribution of this paper is two-fold: we present our solution to all of the above problems and how we integrated them into a single, easy-to-use system<sup>1</sup>. Second, we provide a detailed analysis of each of these components and of the integrated system as a whole. To evaluate the generality of our solution, we carried out extensive experiments on door and drawer opening in five different kitchens and with 29 different articulated objects. We measured a success rate of 51.9% for the whole system. Finally, we share the result of this effort as a well-documented and easy-to-install software package<sup>2</sup> that runs out-of-the-box on any PR2 robot.

<sup>1</sup>[http://www.youtube.com/watch?v=8w1\\_t5zQMnI](http://www.youtube.com/watch?v=8w1_t5zQMnI)

<sup>2</sup>[http://www.ros.org/wiki/pr2\\_cabinet\\_opening](http://www.ros.org/wiki/pr2_cabinet_opening)

## II. RELATED WORK

We review the related work with respect to the above listed contributions.

### A. Handle Detection

There have been various approaches en route to solve the handle perception problem. Klingbeil et al. [2] propose a solution based on a 2D-sliding window object detection algorithm and a supervised learning scheme that produces a dictionary of features consisting of localized templates from cropped training examples. Then a binary classifier is learned using boosted decision trees to select the set of best matching feature matches. In [8] Arisumi et al. propose the identification of knobs and handles using stereo vision and Maximally Stable Extremal Region which requires certain assumptions about the size, height and location of the blob in the image. In contrast, our approach makes use of 3D point cloud clustering in Euclidean space [9] and will be presented in subsection IV-A. A general problem are surfaces with specularity, as specular surfaces are typically not visible in 3D scans. In our approach, we explicitly exploit this fact by specifically searching for handles in regions with missing range values.

### B. Operating Cabinets and Model Learning

Other researchers have focused on the problem of operating doors and drawers using mobile manipulation robots. Many of these approaches make strong assumptions about the kinematics of the articulated object, in particular for opening room doors [2], [3], [10]–[12]. When no kinematic model of the articulated object is available, force- and impedance-controlled methods have proven successful [13], [14]. In our own recent work [7], we demonstrated how articulated objects can be operated using a combination of real-time impedance control and kinematic model learning.

### C. Modelling of Cabinet Appearance

A framework for in-hand-modeling of objects is proposed in [15]. It uses a two step approach for background subtraction, considering the distance to the gripper and searching for components connected to the gripper. Both steps use assumptions not present in our scope of application, where the background is close to and topologically connected with the articulated object. In [16], an approach for the creation of a skin and skeleton model of piecewise rigid articulated objects from 3D point clouds is presented. It is limited in the presence of sliding parallel planes, as the underlying ICP algorithm creates false correspondences. We avoid this by incorporating the learned articulation model in the estimation of the transformation between corresponding points in consecutive frames.

### D. Semantic Maps

We use the KnowRob [17] knowledge processing system to store the articulation models and handle poses as part of the semantic map of our environment. In [18] another system for building the internal representations of space for robots



Fig. 2. Testbed kitchens at the Bosch RTC facility. The upper cabinets are too high and could not be reached by the PR2.



Fig. 3. Testbed kitchen at TUM IAS.

to act in human living environments is proposed. While their system can build conceptual and extensible semantic maps that are useful for the classification of places and scenes, they do not have a possibility to learn and store articulation models for opening cabinets.

### E. The system as a whole

In this paper, we present an integrated system that enables robots to operate cabinets in kitchen environments. The performance of the system as a whole as well as all individual parts were evaluated in five different kitchens, on 29 objects and in more than 100 trials. To the best of our knowledge, no other approach to cabinet opening has been evaluated so extensively. In contrast to our previous work [5] we herein also overcome the assumptions about the thick and rectangular handles which were needed i) for the reliable detection of them, and ii) for the operating of the cabinets based on large overlaps between the surface of the handle and the gripper.

## III. SYSTEM ARCHITECTURE

The individual processing steps of our approach are visualized in Fig. 1. We assume that an initial semantic map of the environment has been built a priori and is available in KnowRob [17]. The map contains information about the environment such as a map for localization, approximate cabinet sizes and poses, rough handle poses, etc. The user starts by specifying a coarse coordinate of a new cabinet to be opened and the robot retrieves the stored handle pose, navigates towards it and starts a detection for it to verify

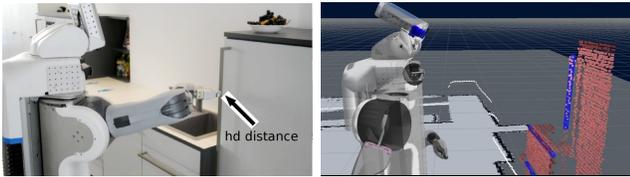


Fig. 4. 3D point cloud-based handle detector. *Left*: handles on cabinets in TUM’s kitchen and  $h_d$  distance between the cabinet face and the handle, *right*: detection of the handles (blue points) in the original 3D point cloud (red points) of the cabinet from the left photo.

the stored handle pose. In case of multiple handles are found, the one closest to the stored handle pose is selected. Subsequently, the robot grasps the handle, switches to an impedance-based cartesian controller and pulls the handle backwards. While the handle moves, the robot continuously estimates the articulation model from its observed toolframe trajectory and uses it to update the equilibrium point fed to the cartesian controller. After the motion comes to a rest, the robot closes the cabinet again by generating cartesian equilibrium points along the learned model. Subsequently, we sample points of the learned model in regular steps and re-execute it with a simple joint space PID controller using inverse kinematics to verify the validity of the learned model. At the same time, we record point clouds of the articulated object in these regular steps that are used for the 3D reconstruction of the articulated bodies.

#### IV. TWO APPROACHES TO HANDLE DETECTION

To enable a reliable and robust manipulation of kitchen cabinets the robot must first recognize and find relevant fixtures, such as handles, which indicate the presence of doors. To find the cabinet candidates in a certain room we first acquire a 3D point cloud model of a room using a Kinect sensor and then apply a sequence of processing steps to classify the previously obtained point cloud into sub-parts such as ceiling, floor, walls and furniture cabinets. Furthermore we segment out the front faces of the cabinets using a RANSAC-based segmentation of planes and thus obtain a vertical region of interest (ROI) which the robot is to explore. The final result of this process is stored into a semantic map as described in [5].

Given the large variation in the appearances and locations of doors and door handles, this is a challenging perception problem. In this paper, we focus on the case of doors with door handles. We identified two types of handle appearances that have different characteristics with respect to sensor data: handles that have specular reflection (Fig. 4) and the ones that do not (Fig. 5). To tackle these two distinct cases we propose a two-fold approach that first tries to recognize and localize a handle in a full 3D model of the given environment. Shall the latter fail we resolve to finding the handle in the parts of the 3D model that lacks range measurements due to the reflection of the sensor’s projected infrared light pattern on specular surfaces.

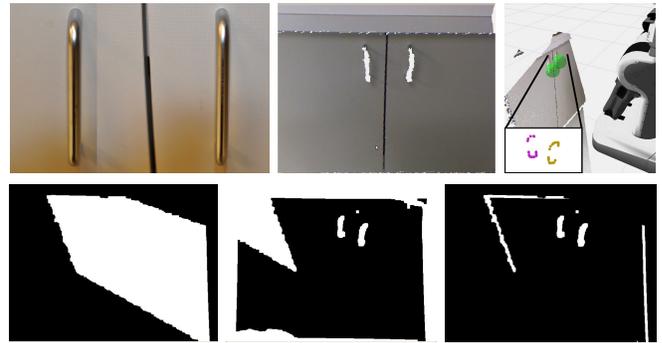


Fig. 5. Visualization of the processing steps for the handle detector based on invalid measurements. *Top-left*: example of two specular handles, *top-middle*: invalid measurements in place of specular handles as seen in a point cloud by Kinect, *top-right*: final handle poses (green spheres) computed from the generated convex hulls visualized in the corner of the subfigure, *bottom-left*: cabinet front face as binary mask (white), *bottom-middle*: invalid measurements as binary mask (white), *bottom-right*: result of a bit-wise conjunction operation.

##### A. Detection of Handles without Specularity

We apply a refined version of the approach proposed by Rusu et al. [9] which uses 3D point cloud data as an input and assumes that the handles are to be found at a certain distance  $h_d$  from a segmented plane of a door along the normal direction of the plane. The parameter  $h_d$  is given by the ADA (Americans with Disabilities Act) requirements as the maximum distance from a door plane where a handle could be located. Their pipeline gets all points whose distance from the plane model is smaller than  $h_d$ , and checks whether their projection on the plane falls inside the bounding polygon of the plane. The actual handle is obtained by first projecting the handle on the door plane, and then fitting the best line within a plane parallel to the door in it using a RANSAC-based line fitting to infer the correct orientation. The geometric centroid of the handle cluster is then used along with the orientation of the line and the plane normal to grasp the handle (see Fig. 4).

##### B. Detection of Handles with Specularity

The handle detector described in the previous subsection works well when the handle is thick enough to be visible in the 3D point clouds acquired by the robot. In practice, we found that this strategy fails when the handles are too thin or specular, so that they are not seen by the sensor. Technically, the Kinect sensor requires a large enough block of the projected pattern to be visible to compute the disparity [19]. In case of thin or specular handles the infrared pattern gets reflected, which results in low correlations during block matching so that the sensor returns missing values in these regions (see Fig. 5, top-middle).

Our key idea is thus to actively exploit patches with invalid measurements (“holes”) in the depth images of the Kinect sensor, as they potentially indicate the presence of a specular handle. To robustly detect these holes and restore their poses we proceed as follows. Firstly we create two binary masks (2D images) of the invalid measurements and projected points of the detected cabinet face (ROI). Next we perform

a series of dilation and erosion operations on such generated images to fill the holes in the binary images. Following we perform a bit-wise conjunction of the two binary masks and obtain an image containing handle candidates within the ROI (Fig. 5, bottom row). Following we apply an Euclidean clustering using a region growing approach and keep only clusters  $C$  that correspond to the expected size of the handle in the image. To restore the position of the centroid of the handle we first compute the convex hull  $H$  around every cluster  $c_i$  from a set of clusters  $C$ , find the corresponding 3D point in the ROI point cloud for every point  $h_i$  on the hull  $H$  and compute the centroid (Fig. 5, top-right). The orientation of the handle is calculated by first converting the image with the clusters into an edge image using a Canny operator and then using a RANSAC-based line fitting. The pose of the handle is finally transformed into the coordinate frame of the base of the robot (with  $Z$  pointing upwards and  $X$  pointing forwards) and the handle is grasped. To find the distance between the handle and the supporting plane we make use of the PR2 tactile sensors in its fingertips: We steer the robot’s arm towards the transformed pose and the  $X$  component of the handle position is determined as the contact point between the handle and fingertip.

## V. OPERATING ARTICULATED OBJECTS

After the robot has firmly grasped the handle, it starts to operate the articulated object using an impedance-based controller. The controller allows us to specify in each time step  $i$  a Cartesian equilibrium pose  $\mathbf{x}_i^{CEP} \in SE(3)$  specifying both the desired 3D position and 3D orientation that the robot tries to reach with its gripper in the absence of any external force.

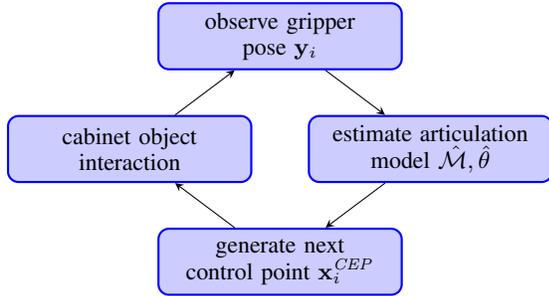


Fig. 6. Overall control structure. The robot iteratively estimates the kinematic model of the articulated object from the perceived trajectory of its end effector and evaluates it to generate the next Cartesian equilibrium point.

The schematic overview of our approach is depicted in Fig. 6. Initially, the robot gently pulls the handle backwards by moving the Cartesian equilibrium point towards the robot. As soon as the motion starts, the robot records the trajectory of its gripper  $\mathbf{y}_{1:n}$  with  $\mathbf{y}_i \in SE(3)$  feeding the joint encoder values through forward kinematics. From this partial trajectory, it continuously (re-)estimates the kinematic model  $\mathcal{M} \in \{\text{rigid, prismatic, rotational}\}$  and model-specific parameter vector  $\theta \in \mathbb{R}^d$  (encoding radius, rotation axis, etc.)

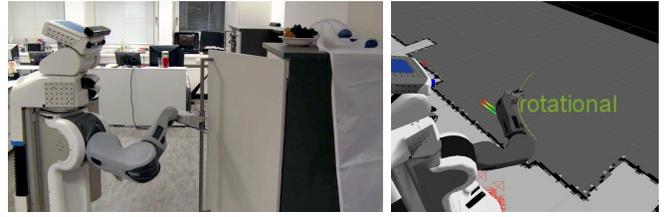


Fig. 7. TUM’s PR2 robot operates a fridge, learning its kinematic model.

of the articulated object:

$$\hat{\mathcal{M}}, \hat{\theta} = \arg \max_{\mathcal{M}, \theta} p(\mathcal{M}, \theta \mid \mathbf{y}_{1:n}). \quad (1)$$

To solve (1), we first fit the parameter vector of all model candidates using an maximum-likelihood sample consensus (MLE-SAC) estimator and then select the best model according to the Bayesian information criterion (BIC). In turn, the robot uses this model to predict the continuation of the trajectory and to generate the next Cartesian equilibrium point  $\mathbf{x}_{n+1}^{CEP}$ . After the motion of the end effector has come to a rest, we determine the observed configuration range, i.e., the opening angle for doors and opening distance for drawers of the articulated object. In sum, this gives us the full kinematic model of the articulated object. Figure 7 shows an example where the robot operates a fridge (left image) using this approach and learns its kinematic model (visualized in the right image). More details about our approach can be found in [7].

### A. Using Learned Articulation Models

1) *Re-opening and closing:* After the robot has successfully opened and closed an articulated object for the first time using the controller described in the previous subsection, we have an estimate of the kinematic model  $\hat{\mathcal{M}}, \hat{\theta}$  that we can use for subsequent interactions. We sample a noise-free trajectory  $\bar{\mathbf{x}}_{1:n}$  from the model over the configuration range of the articulated object to be used in further interactions using normal position control. By checking a slightly extrapolated trajectory against the robot’s workspace, we can tell if the end of motion due to a workspace limit. We then query an inverse reachability map to generate a robot pose that brings more of the extrapolated trajectory into the robots reach, and retry.

Similarly, we can use the inverse reachability map and search for base poses that not only allow for the full opening of the cabinet but also to bring the inside of the cabinet into the workspace of the other hand, for example for retrieving objects from the fridge.

2) *3D reconstruction of the articulated objects:* The noise-free trajectory  $\bar{\mathbf{x}}_{1:n}$  is used again during 3D reconstruction of the articulated parts. In principle, the observed gripper poses could be employed, but they are – partly due to gripper slippage – too noisy and thus degrade the reconstruction process.

While re-opening the articulated object along the re-sampled trajectory  $\mathbf{x}_i \in SE(3)$  for  $i = 1, \dots, n$ , the robot observes a point cloud  $C_i \subset \mathbb{R}^3$  from the head-mounted

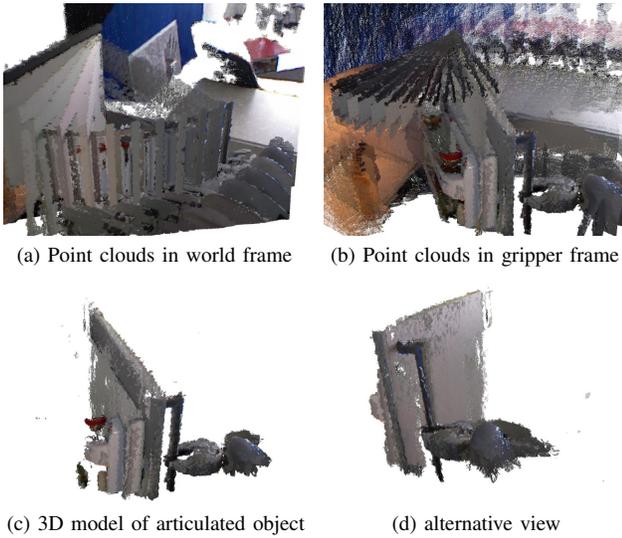


Fig. 8. (a) and (b): The summed sets of point clouds reprojected in the world and gripper frame. Reconstructed 3D models of the object (c),(d).

Kinect sensor for each opening angle (doors) or distance (drawers). Our goal is to assign each of the points in  $C_i$  either to the background or to the moving parts of the articulated object. We use a maximum-Likelihood classifier to separate both classes of points. In particular, we compute the likelihood of a point  $u$  to belong to the set of all point clouds that are static with respect to the world coordinate frame (Fig. 8a):

$$C_{\text{world}} = C_1, \dots, C_n, \quad (2)$$

or to the set of point clouds expressed in a coordinate frame relative to the moving gripper (Fig. 8b), i.e.,

$$C_{\text{gripper}} = \mathbf{x}_1^{-1}C_1, \dots, \mathbf{x}_n^{-1}C_n, \quad (3)$$

where  $\mathbf{x}_1^{-1}$  is an inverse of a gripper pose in the world coordinate frame. We define the observation likelihood of a point  $u \in \mathbb{R}^3$  to belong to such a set of point clouds  $C$  using Gaussian radial basis functions, i.e.,

$$p(u | C_1, \dots, C_n) \propto \sum_{v \in C_i} \exp\left(-\left(\frac{v-u}{\sigma}\right)^2\right), \quad (4)$$

where  $\sigma$  refers to the observation noise of the sensor and  $v$  is the point closest to  $u$  in  $C_i$ . With this, we can, using a maximum-likelihood classifier, assign each point to either the static or the moving part of the scene. In more detail, we assign a point  $u$  observed from a static camera while the gripper was in a pose  $\mathbf{x}$  to the articulated part the following condition is fulfilled:

$$p(\mathbf{x}^{-1}u | C_{\text{gripper}}) > p(u | C_{\text{world}}). \quad (5)$$

Finally, to suppress noise, we exclude points with low support, i.e., points where both  $p(\mathbf{x}^{-1}u | C_{\text{gripper}})$  and  $p(u | C_{\text{world}})$  have a likelihood lower than an empirically found fixed threshold  $\tau$ .

An example of the reconstructed 3D model of a fridge door is given in Fig. 8. The robot observed 12 point clouds

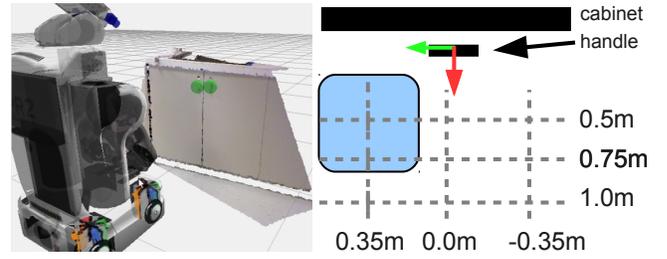


Fig. 9. Left: PR2 during handle detection test, right: grid of 9 observation vertices for the handle detection test. From every vertex three measurements per handle were taken.

of the door while opening it with its arm. Fig. 8c shows the 3D model of the segmented fridge door. Note that in our current implementation, the gripper is not filtered out and thus also part of the articulated object. A readily available self-filter can be used to filter the arm out of the point clouds before reconstruction.

The expected benefits of reconstructing the 3D model of the object are manifold: first, the model enables the robot to estimate and verify the opening angle or distance of the cabinet later, such as when it was moved by a human, by comparing observed sensor data to the learned appearance model. Second, it allows the robot to plan collision-free opening trajectories, and third, it can be used to calculate additional grasp poses apart from the handles, that could be used to facilitate opening the cabinets in constrained spaces.

## VI. RESULTS AND DISCUSSION

To validate our approach, we evaluated both the individual components as well as the overall performance of the system in four kitchens at Bosch RTC (See Fig. 2) and one kitchen at TUM (See Fig. 3).

Starting from a user-provided cabinet from the KnowRob system and the choice of a handle detector, the robot navigated to the predefined distance with respect to the stored handle pose and brought the handle into the viewing frustum of the sensor. Next, it detected the handle and brought its base into a manipulation pose. To maximize available force, the configuration of the arm was chosen such that that the strongest joint – in case of the PR2 the shoulder panning joint – rotated back when pulling the handle towards the robot. After the handle was firmly grasped, the robot started to open the cabinet using the impedance controller while concurrently learning the articulation model. In case of a success, the robot closed the cabinet and re-opened it again using the re-sampled trajectory with a jointspace controller after applying inverse kinematics while also recording point clouds for the 3D reconstruction.

### A. Detection and Localization of Handles

Before we evaluated the whole system, we carried out an experiment for measuring the reliability and accuracy of our handle detection module. To account for different distances and viewing angles two handles were observed from nine sample points — vertices in the grid as depicted in Fig. 9. To achieve statistical significance every test was repeated three

	positive	negative
positive	45	4
negative	5	0

TABLE I. Confusion matrix for the evaluation of the specular handle detection module. Over a rectangular grid of 0.7 m  $\times$  0.5 m, the true positive rate was 83.3%.

times per handle with slight variations which resulted in 54 total measurements. We visually validated the correctness of the measurements and present the results in Table I.

The two major sources of false positive measurements are i) the presence of kitchen appliances with metal surfaces (dishwashers, refrigerators) and ii) the limitation of the sensor’s minimum range (0.5 m). Since in both cases invalid measurements are returned by the sensor we occasionally encounter situations where the handle size and position hypotheses match to those observed for real handles. The measurements were however irregular which would allow us to filter them out using statistical sampling methods. The reason for false negative measurements is in that under certain views the projected infrared pattern does not get reflected which results in valid range measurements. Note that this case can be easily overcome by using both of herein proposed handle detectors together.

### B. Evaluation of the whole system

cabinet type	number of cabinets	number of trials
drawer	13	41
door	16	63
sum	29	104

TABLE II. Overview of the number of cabinets per class and the number of trials carried out in our quantitative evaluation.

After this preliminary experiment, we evaluated the system as a whole on all 29 cabinets located in the five kitchens of Bosch and TUM (see Table II). In total, we ran 104 trials on these 29 cabinets. The results of the experiment are given in Table III.

After a handle was successfully detected (in 93.3 %), the controller presented in Section V moved the mechanisms in 79.2 % of the trials (see Table IV). We consider opening a cabinet full success, if the robot could open it further than 0.4 m for a drawer or more than 60 degrees for a door. By this measure, the PR2 only succeeded in 56.2 % of all trials to open the cabinet far enough.

activity	trials	successes	success rate
handle detection	104	97	93.3 %
operate cabinet and learn model	97	54	56.2 %
re-execute learned model	54	54	100 %
overall performance	104	54	51.9 %

TABLE III. Experimental results of the whole-system evaluation. Handle detection and re-execution of a learned model work have high success rates, but operating novel cabinets only succeeds in 56.2 %. See Table IV for an analysis of failure modes.

type of failure	trials	failed trials	rate
(a) handle detection	104	7	6.7 %
(b) slippage of gripper	97	6	6.2 %
(c) robot too weak at start	97	6	6.2 %
(d) robot too weak during motion	97	23	23.7 %
(e) overall failures during control	97	35	36.1 %
(f) overall failures in learning phase	97	43	44.3 %
(g) sum (a)+(f)	104	50	48.1 %

TABLE IV. Failure modes when articulating novel cabinets. The most prominent failure mode is that the PR2 is not strong enough to open the cabinet (6.2 % and 23.7 %). Note that (c), (d) and (f) overlap. Furthermore (g) includes (e) and additional failures in model estimation.

In 20.8 % of all trials, the movement failed from the beginning, got stalled in the middle, or the gripper slipped off. In six cases, the cabinet did not open at all: we found that it is virtually impossible for the PR2 to open the very heavy large drawers (see Fig. 3: containers 101 and 105, all 6 trials) in the TUM kitchen - they require the extraordinary amount of about 50N of initial force to open. This is due to them being equipped with 4 self-retraction springs instead of the usual one. In teleoperation, we were able to open them by power-grasping their handles instead of precision/pinch-grasping them and then driving the robot’s base backwards with a stretched arm . We do not intend to integrate this mechanically stressful behaviour into the robots’ autonomous skills, but instead hope that future generations of mobile manipulation robots will be strong enough to operate such furniture.

In seven cases opening a dishwasher failed. The robot got stalled and could not pull further after opening the cabinet only slightly. The respective dishwasher doors are heavy and have a high stiffness. The PR2 is also in this case limited by its hardware. Bi-manual operation could be a solution we will investigate further, although preliminary bi-manual experiments using the tele-operation did not give promising results. The cases where the gripper slipped off the handles showed that we possibly overtuned the maximum force and acceleration parameters of the underlying Cartesian impedance controller. After successfully grasping the handle, it slipped off with a very dynamic motion of the arm. To solve this, we consider developing a stepped approach, first trying with a more conservative controller setup, and only trying with maximum force when necessary and thus avoiding too harsh accelerations.

### C. Learning Rate and Accuracy

We evaluated the learning rate as a function of the trajectory length during opening the containers. Figure 10 gives the result. We found that the drawers require approximately 6 cm and the doors approximately 25 cm to be recognized correctly with a probability of above 95%. After the motion came to a halt, we evaluated the opening distances and estimated opening radii during model learning. Interestingly, the model almost always underestimates the radius of doors (on average by -16.7% with a single standard deviation of 12.1%), while drawer opening distances were generally

cabinet type	trials	translational err.	rotational err.
drawers	18	0.664cm	4.96°
doors	35	0.516cm	36.2°
average	53	0.567cm	25.6°

TABLE V. Trajectory reprojection errors. For doors, the difference in observed and expected gripper orientation indicates significant slippage.

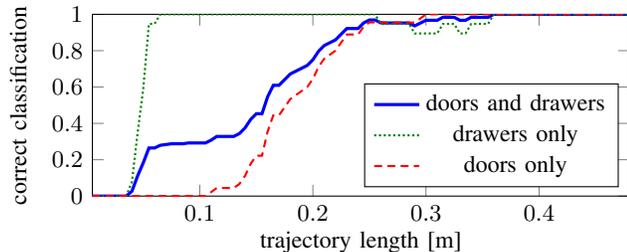


Fig. 10. Learning curve over all trials. When the robot achieves to open a container farther than 25 cm, the success rate of classifying it correctly (as either door or drawer) is above 95%.

overestimated (on average +6.9% with a standard deviation of 18.0%). In both cases, we believe that this is the result of the high dynamics and the resulting slippage of the gripper during articulation. This hypothesis can also be confirmed when the observed trajectories and trajectories resampled from the learned models are compared. The result of this comparison is given in Table V. In the case of the doors, the rotational errors are surprisingly high (0.5 cm and 36.2°). This is most likely the result of a slippage due to the inherent behavior of our continuous model estimation. As the model complexity of the prismatic model is lower, the BIC prefers the prismatic model during model selection over the rotational model as long as this is not in contradiction with the observed trajectory. This means that our controller follows a line segment instead of a circle for the first few centimeters, and thus, keeps the gripper orientation constant. Large errors therefore occur between the observed gripper pose and the gripper pose predicted by the learned model, especially on round thin handles that easily slip rotationally in the PR2’s parallel gripper. In contrast, the differences of the observed gripper poses and the model predictions for drawers are relatively low (0.6 cm and 5°).

## VII. CONCLUSIONS AND FUTURE WORK

In this paper, we presented our generalized framework for operating cabinets in kitchen environments. We contribute a novel detector for specular handles, and evaluated it in a large set of different poses. We developed a compliant controller that learns the kinematic model of an articulated object while operating it. The learned model enables the robot to operate the cabinet more robustly. Subsequently, we reconstruct a 3D model of the moving parts from different views of the articulated object. Finally, we evaluated our framework on a large set of different cabinets located in five kitchens spread over two research institutions. In our experiments, we found that our framework enabled two PR2 robots to operate more than 50 % of all cabinets successfully. By studying the failure

modes, we found that the PR2 is in many cases mechanically too weak to operate all cabinets found in typical kitchens. Yet, for many relevant robotic applications – such as setting the table or tidying up – this subset of robustly operable cabinets might already be sufficient.

## REFERENCES

- [1] A. Jain and C. Kemp, “Behavior-based door opening with equilibrium point control,” in *Proc. of the RSS Workshop on Mobile Manipulation in Human Environments*, Seattle, WA, USA, 2009.
- [2] E. Klingbeil, A. Saxena, and A. Y. Ng, “Learning to open new doors,” in *IEEE/RSS Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2010.
- [3] W. Meeussen, M. Wise, S. Glaser, S. Chitta, C. McGann, M. Patrick, E. Marder-Eppstein, M. Muja, V. Eruhimov, T. Foote, J. Hsu, R. Rusu, B. Marthi, G. Bradski, K. Konolige, B. Gerkey, and E. Berger, “Autonomous door opening and plugging in with a personal robot,” in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Anchorage, AK, USA, 2010.
- [4] S. Wieland, D. Gonzalez-Aguirre, N. Vahrenkamp, T. Asfour, and R. Dillmann, “Combining force and visual feedback for physical interaction tasks in humanoid robots,” in *Proc. of IEEE-RAS Intl. Conf. on Humanoid Robots (Humanoids)*, Paris, France, 2009.
- [5] N. Blodow, L. C. Goron, Z.-C. Marton, D. Pangercic, T. Rühr, M. Tenorth, and M. Beetz, “Autonomous semantic mapping for robots performing everyday manipulation tasks in kitchen environments,” in *IEEE/RSS Intl. Conf. on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, September, 25–30 2011.
- [6] J. Sturm, A. Jain, C. Stachniss, C. Kemp, and W. Burgard, “Operating articulated objects based on experience,” in *Proc. of the IEEE/RSS Intl. Conf. on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, 2010.
- [7] J. Sturm, C. Stachniss, and W. Burgard, “Learning kinematic models for articulated objects,” *Journal on Artificial Intelligence Research (JAIR)*, vol. 41, 2011.
- [8] H. Arisumi, N. Kwak, and K. Yokoi, “Systematic touch scheme for a humanoid robot to grasp a door knob,” in *ICRA*, 2011, pp. 3324–3331.
- [9] R. B. Rusu, W. Meeussen, S. Chitta, and M. Beetz, “Laser-based perception for door and handle identification,” in *Intl. Conf. on Advanced Robotics (ICAR)*, Munich, Germany, 06/2009 2009.
- [10] A. Petrovskaya and A. Ng, “Probabilistic mobile manipulation in dynamic environments, with application to opening doors,” in *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, Hyderabad, India, 2007.
- [11] S. Srinivasa, D. Ferguson, C. Helfrich, D. Berenson, A. C. Romea, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and J. M. Vandeweghe, “HERB: a home exploring robotic butler,” *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, January 2010.
- [12] S. Chitta, B. Cohen, and M. Likhachev, “Planning for autonomous door opening with a mobile manipulator,” in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Anchorage, AK, USA, 2010.
- [13] G. Niemeyer and J.-J. Slotine, “A simple strategy for opening an unknown door,” in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Albuquerque, NM, USA, 1997.
- [14] A. Jain and C. Kemp, “Pulling open doors and drawers: Coordinating an omni-directional base and a compliant arm with equilibrium point control,” in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Anchorage, AK, USA, 2010.
- [15] M. Krainin, P. Henry, X. Ren, and D. Fox, “Manipulator and object tracking for in-hand 3d object modeling,” *International Journal on Robotics Research (IJRR)*, 2011.
- [16] Y. Pekelny and C. Gotsman, “Articulated object reconstruction and markerless motion capture from depth video,” *Computer Graphics Forum*, vol. 27, no. 2, pp. 399–408, 2008.
- [17] M. Tenorth and M. Beetz, “KnowRob — Knowledge Processing for Autonomous Personal Robots,” in *IEEE/RSS Intl. Conf. on Intelligent Robots and Systems*, 2009, pp. 4261–4266.
- [18] A. Pronobis, “Semantic mapping with mobile robots,” Ph.D. dissertation, Royal Institute of Technology (KTH), Stockholm, Sweden, June 2011. [Online]. Available: <http://www.pronobis.pro/phd>
- [19] *Technical description of Kinect calibration*. [Online]. Available: [http://www.ros.org/wiki/kinect\\_calibration/technical](http://www.ros.org/wiki/kinect_calibration/technical)