Integration of an Autonomous System with Human-in-the-Loop for Grasping an Unreachable Object in the Domestic Environment

Jaeseok Kim, Raffaele Limosani and Filippo Cavallo
The Biorobotics Institute, Sant'Anna School of Advanced Studies, Viale Rinaldo Piaggio, Pontedera, Pisa, Italy

Keywords: Mobile Manipulation, Autonomous System with Human-in-the-Loop, Assistive Robotics, Grasping an Unreachable Object, Service Robotics.

Abstract: In recent years, autonomous robots have proven capable of solving tasks in complex environments. In particular, robot manipulations in activities of daily living (ADL) for service robots have been widely developed. However, manipulations of grasping an unreachable object in domestic environments still present difficulty. To perform those applications better, we developed an autonomous system with human-in-the-loop that combined the cognitive skills of a human operator with autonomous robot behaviors. In this work, we present techniques for integration the system for assistive mobile manipulation and new strategies to support users in the domestic environment. We demonstrate that the robot can grasp multiple objects with random size at known and unknown table heights. Specifically, we developed three strategies for manipulation. We also demonstrated these strategies using two intuitive interfaces, a visual interface in rviz and a voice user interface with speech recognition. Moreover, the robot can select strategies automatically in random scenarios, which make the robot intelligent and able to make decisions independently in the environment. We demonstrated that our robot shows the capabilities for employment in domestic environments to perform actual tasks.

1 INTRODUCTION

In the home environment, perception is used to recognize a variety of objects; however, a service robot might not be able to detect all of the objects in every circumstance. In other words, when the robot attempts to recognize multiple objects on a table, it is difficult to detect all objects because many variables such as the shape of object, robot hand position and etc. should be considered. However, if a human can support the judgment of the robot, the robot can acquire the specific object needed quite easily.

To find solutions for performing tasks in the home environment, many types of service robots have been developed. In particular, robots that provide fully autonomous system and activities of daily living (ADL) for older people have been developed. One representative service robot is Care-O-bot, which was developed with basic technologies for delivery, navigation, and monitoring for users (Schraft et al., 1998). Moreover, in recent years, several projects have featured different robots that integrate smart home technology for healthcare, shopping, garaging (Cavallo et al., 2014), and communication with users by gesture and speech (Torta et al., 2012).

Despite enhanced functionalities of the service robots, we still face several challenges of ADL in the domestic environment. Particularly with tasks such as grasping objects iteratively in the environments, the capabilities of current robots are still lacking. To solve these problems, robots typically focus on either a fully autonomous or a fully teleoperated system. However, many limitations with perception and manipulation remain. A possible solution to overcome these issues is to use autonomous system with human-in-the-loop, in which the human operator controls the robot in a remote site with a high level of abstraction.

In this paper, The development of the integration of an autonomous system with human-in-the-loop was presented for grasping an unreachable object in the domestic environment. The main contributions of this paper are the following:

- Multi-object segmentation was implemented and it supports grasping point detection that used grasping the object with two different grasp poses using a depth camera.
- Three mobile manipulation strategies were developed for picking and placing unreachable objects with various and unknown table heights.

Kim, J., Limosani, R. and Cavallo, F.
Integration of an Autonomous System with Human-in-the-Loop for Grasping an Unreachable Object in the Domestic Environment.
DOI: 10.5220/0007839503060315
In *Proceedings of the 16th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2019)*, pages 306-315
ISBN: 978-989-758-380-3

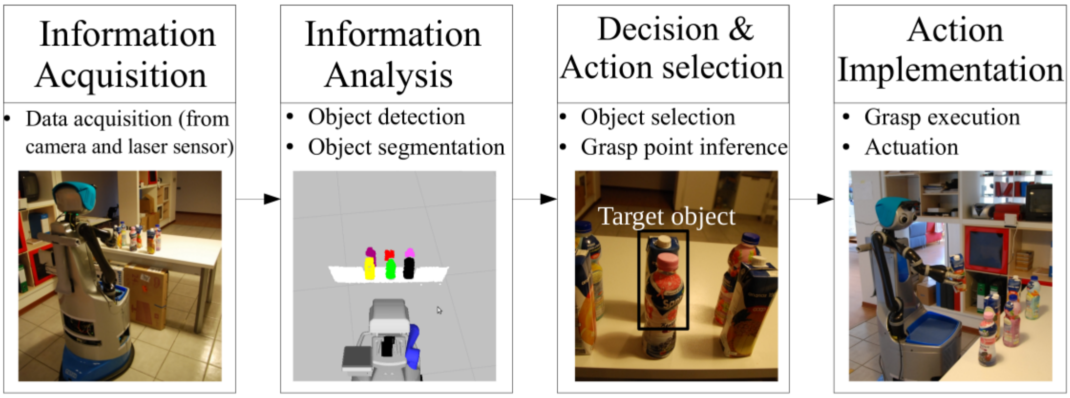

Figure 1: The mobile manipulation task is operated by human capabilities applied with Sheridan's four-stage model (a) Information acquisition, (b) Information analysis, (c) Detection and action selection, and (d) action implementation.

- Perception, visual and voice interfaces, and motion planning were integrated as functionalities of grasping an unreachable object system framework.

2 RELATED WORK

Mobile manipulation tasks for grasping an object in the domestic environment have been studied extensively over decades (Dogar and Srinivasa, 2011; Kitaev et al., 2015; Stilman et al., 2007; Özgür and Akın, ; Fromm and Birk, 2016).

Grasping objects are frequently used for autonomous manipulation system in the domestic environment. However, grasping unreachable objects (occluded objects) are still one of the issues in the environment. In order to solve the issue, the pushing manipulation systems for grasping unreachable an object have been developed. Dogar *et al.* (Dogar and Srinivasa, 2011) suggest a framework to generate a sequence of pushing actions to manipulate a target object. However, the pushing action system needs adequate space in which to shift or remove objects. To solve the problem, a sequence of manipulation actions to grasp the objects is suggested. Stilman *et al.* (Stilman et al., 2007) use a sampling-based planner to take away the blocking objects in a simulation. Moreover, Fromm *et al.* (Fromm and Birk, 2016) propose a method to plan strategies for a sequence of manipulation actions. Based on the previous paper, a sequence of manipulation actions for grasping unreachable objects was considered. However, the discussed literatures show that the robot already knows the target object before the robot manipulation starts. To overcome the object selection during the robot manipulation, user-interfaces operated by autonomous system with human-in-the-loop are developed.

Many prior works address that user-interfaces are developed for object selection in an autonomous system with human-in-the-loop. For remote selection of an object by people at home, the graphical point-click interface system was developed (Pitzer et al., 2011). The interface is allowed to drag, translate, and rotate to select a target object by a person. In addition, an interface is used to generate waypoints for desired gripper position to conduct grasping tasks (Leeper et al., 2012). The interface systems support the object selection problem using human capabilities. However, the interface systems on the papers only consider grasping reachable objects, which are not occluded, and simple task planning is applied. In addition, to select the target object, a human operator should concentrate on the visual display and take their time when selecting an object. For theses reasons, we developed three mobile manipulation strategies for grasping occluded and different grasp poses to support grasp planning with intuitively user interfaces for object selection.

3 SYSTEM ARCHITECTURE

The goal of our work is to develop a robotic system that will be able to help people in ADL. In particular, we studied the scenario in which a user needs a particular object located on a table and asks the assistant robot to find and bring the object.

In fact, the preferred way to achieve the scenario is to operate the robotic system automatically. However, the capability to recognize a target object, calculate eligible grasp poses, and generate task planning for complex tasks are still being researched. The mobile manipulation task, which is part of the robotic system, should be automated. Parasuraman *et al.* (Parasuraman et al., 2000) proposed a model for different levels of automation that provides a framework made

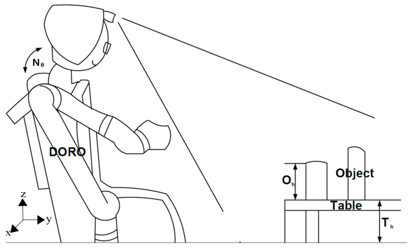

Figure 2: A simple pictorial diagram with variables such as neck angle(N_θ), table height(T_h) and object height(O_h). Look at the text for the variables applied in the equations.

up of four classes: (1) information acquisition, (2) information analysis, (3) decision and action selection, and (4) action implementation. In current work, we adapted the same framework in our robotic system (see Figure 1).

4 AUTONOMOUS SYSTEM WITH HUMAN-IN-THE-LOOP DESCRIPTION

Based on Sheridan's four-stage model, which was previously described, *Decision & Action selection* were conducted as a good starting point for the autonomous system with human-in-the-loop concept. We aimed to develop a system that is operated by minimum human effort. The human operator contributes by interpreting the environment from the camera images and by choosing a low level of automation in the third stage of Sheridan's model.

The final objective of the mobile manipulation task is to pick up an object on a table of unknown height and bring it to a user. In our system, as the robot finds a table, it measures the table height and adjusts its neck angle accordingly. Then, robot detects and segments multiple objects that are positioned in two rows (front and back). As the robot completes extraction of the objects with several RGB colors, the user selects a row and a target object using voice and visual interfaces. Based on the table height information, and row and object information (height, length, weight, and distance), the robot selects one of three mobile manipulation strategies that we have developed for grasping an object in the back row. The strategies were designed with different grasp poses according to the table heights.

For grasping an object, we followed two scenarios, In the first scenario, the user employed a known table height fixed at 70, 80, 90, or 100 cm. Moreover, in the second scenario, the user employed an unknown table height which is measured by robot itself, and robot can decide empirically which strategies are better for grasping. From several experimental trials, empirical results suggest three strategy modes for better grasping:

$$\begin{cases} 1 & T_h < 77cm \\ 2 & T_h < 87cm \\ 3 & T_h < 100cm \end{cases} \tag{1}$$

where T_h is table height measured by camera. For grasping objects and controlling the arm, we used Point Cloud Library (PCL) (Rusu and Cousins, 2011) and Moveit (Chitta et al., 2012) library.

5 IMPLEMENTATION OF AUTONOMOUS SYSTEM WITH HUMAN-IN-THE-LOOP SYSTEM

The goal of the autonomous system with human-in-the-loop is to provide support to improve the quality of human life. Thus, we considered grasping an object from a table of unknown height in the domestic environment, which schematically is shown in Figure 2 with our robot. Actually, many studies for grasping objects (Dogar and Srinivasa, 2011; Kitaev et al., 2015; Fromm and Birk, 2016) were tested using fixed table height and viewpoint. However, if the viewpoint is changed, detection of objects on the table will be difficult by the robot. Therefore, to overcome the difficulty of detecting objects from a different viewpoint, we adjusted the neck angle of the robot based on table height. Before applying the fixed neck angle, the robot needs to find a table. Thus, the initial neck angle was set at the lowest position to find a lower table height. After the table was segmented by PCL, a point (which is calculated by averaging all the coordinate points on the top surface of the table) was substracted from the base frame of the robot (see Figure 4), and only the z-axis value was used to calculate table height. Then the value was stored for changing the neck angle and choosing the strategies. Next, the neck angle of the robot was adjusted by the interpolation method. To interpolate the neck angle, we set the maximum and minimum range of neck angle and table height. Moreover, the linearly interpolated neck angle helped the robot to detect multiple objects easily.

$$N_{\theta,d} = N_{\theta,min} + (T_h - T_{h,min}) \frac{N_{\theta,max} - N_{\theta,min}}{T_{h,max} - T_{h,min}} \tag{2}$$

where $N_{\theta,d}$ is the desired neck angle, $N_{\theta,max}$ and $N_{\theta,min}$ are maximum and minimum neck angle, respectively,

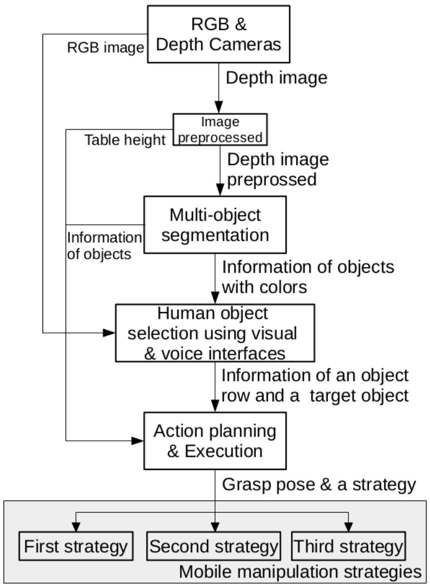

Figure 3: The flow chart of the autonomous system with human-in-the-loop; It is included with image preprocessing, multi-object segmentation, object selection and action planning with developed strategies.

and $T_{h,max}$ and $T_{h,min}$ are maximum and minimum table heights, respectively. The T_h is the current table height described in Figure 2. To establish an appropriate area for grasping an object, the robot secures the workspace using a laser sensor to measure the distance between the table and robot base. After the robot judges that the workspace is appropriate for the manipulation, it starts performing task given. This process is represented in the flow chart in Figure 3.

5.1 Multi-object Segmentation

In our work, we adapted and modified the approach of Trevor*et al.* (Trevor et al., 2013) to segment multiple objects by an organized point cloud library. We used a depth camera (Xtion) for acquiring depth data instead of an RGB camera to increase depth data accuracy. For each point P(x,y), a label L(x, y) is assigned. Points belonging to the same segment will be assigned to the same label based on the Euclidean clustering comparison function (see (Trevor et al., 2013) for more details). To segment the objects accurately, some of the large segments, like the plane surface, will be excluded. In addition, if the distance between the two points in the same label set is more than a threshold, one of the points will be discarded because of increasing object segmentation speed. The area to be clustered and the threshold of points for each object. The points of each object clustered between 1500 to 10000 points were chosen experimentally. To

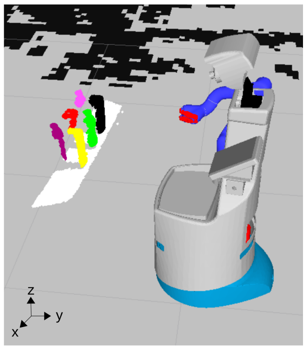

Figure 4: Real-time multi-object segmentation on the table visualized in rviz (Hershberger et al., 2011)

distinguish between multiple objects easily, the object were covered with six RGB colors. The result of this segmentation process is presented in Figure 4. This process is described in the flow chart shown in Figure 3).

5.2 Human Object Selection

For our robotic platform, object selection was done in two ways: 1) voice and 2) visual. We also believe that a combination of these two methods could be easily accessible for very old people who cannot move. Moreover, our interface platform includes a tablet for the voice user interface (see Figure 5(b)) and rviz in a PC for the visualization interface (see Figure 5(a)). The visualization interface, which includes RGB colors and depth information of the environment, was provided for the selection system; the voice user interface is based on speech recognition (Sak et al., 2015). The selection system consisted of three steps:

- 1. Select one of the rows of multiple objects
- 2. Choose an object desired in the same row
- 3. Choose an object desired in a different row

First, the user selects the object from the front or back row of the table. After identification of the row (front or back), the target object selection is done (see Figure 3).

5.3 Action Planning & Execution

Among multiple objects, grasping an object is still a challenge. Thus, we tried to find a grasping point with simply shaped objects such as a bottle or box, which are common household objects in the domestic environment. In addition, the grasping point was used to generate possible hand poses relative to the object for action planning.

To extract the grasping point from each object, we used the 3D centroid function in PCL and configured

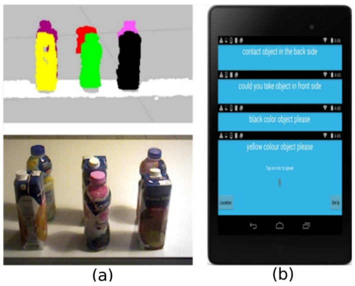

Figure 5: (a) Visualization interface system, (b) Voice user interface system.

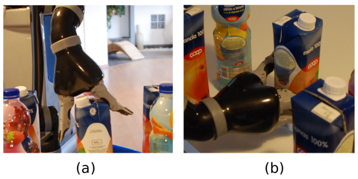

Figure 6: (a) Top grasp pose, (b) Side grasp pose.

the grasp poses. In our case, we characterized two types of grasp poses:

•Top pose: It is aligned by the robot hand to the object in the vertical plane (along the x- and z-axis), and opening and closing of the robotic hand is in the direction of the x- or y-axis (see Figure 6(a)).

•Side pose: It is defined in the horizontal plane (along the x- and y-axis), and the opening and closing direction of the robotic hand is the same as previous (see Figure 6(b)).

To grasp the object, we used the motion planning library, which includes capability for collision avoidance, self-collisions, and joint limit avoidance of the robot arm in the domestic environment. The motion planning library (Moveit) was used for executing three mobile manipulation strategies. During the motion planning, the position of the robot hand plays an important role in grasping. For this reason, pre-grasp position (it is an offset from the target object with the two grasp poses) was developed. After the pre-grasp position was obtained, the palm of the robot hand approached the surface of the target object to grasp it. Based on these technologies, the strategies were enhanced to avoid crashes between the robot arm and robot body during the operation (Ciocarlie et al., 2014) (see Figure 3).

5.4 Developed Mobile Manipulation Strategies

Three strategies of the mobile manipulation were conceived to grasp an object, which was apart from the robot. A set of 6 objects, arranged in two rows, was placed in front of the robot (see Figure 5(a)). We consider grasping objects placed in the back row because grasping front row objects are an easy task that we have already developed. Before starting the strategies for grasping an object, we need to accomplish three steps. The first step is initialization of the robot arm. The next step is to transform the coordinates of multiple objects from camera frame to robot base frame for manipulation. The last step is pre-grasp position based on table height. These three steps are described in Algorithm 1 (lines 2 to 5). Actually, these steps are capable of grasping an object on a table, but grasping back-row objects always fails due to the obstruction caused the front-row objects. For these reasons, we developed three strategies for the mobile manipulation for grasping an object in the back row.

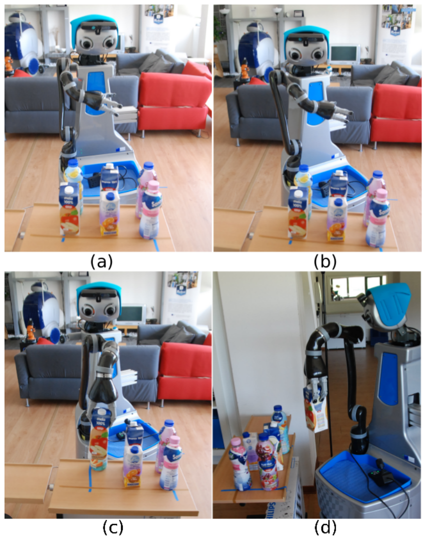

Figure 7: The first strategy for manipulation: (a) The robot moves close to the table, (b) The mobile platform is rotated to grasp the target object. (c) Top grasp pose is implemented to grasp the object directly. (d) After grasping the object, the robot arm returns back to the initial position.

• **The First Strategy.**
The objective of the **first strategy** was to grasp an object on the approximately 70cm high table, directly from the back row, to reduce manipulation time. The mobile platform was pre-defined to be at a rotated angle and also the specific neck angle that supports segmentation of objects in the back row was set. Actually, when the same sizes of objects are detected, the visualization of the object size shows differently because the distance from the camera to each object is different. In addition, the objects in the front row would be obstructed during grasping, and it will be difficult to detect the entire size of the back row objects. For this reason, the function of the linear interpolation (the same as Equation 2) with different variables was developed. The output of the interpolation

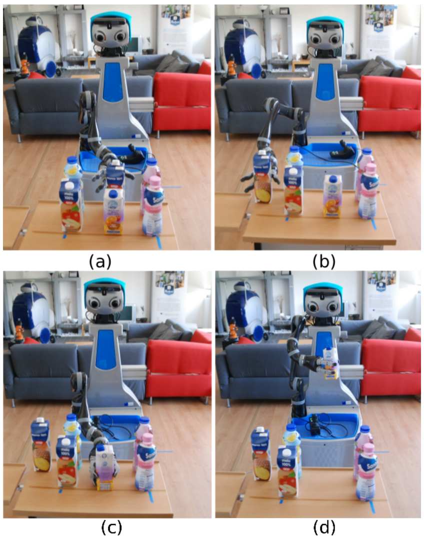

Figure 8: The second strategy shows the robot performing lateral grasp to pick up an object in the back. (a) First, the object in front of the object selected is removed. (b) The object is placed on the empty place. (c) The robot grasps the target object with a side grasp pose. (d) As the object grasps, the arm starts to return back to the initial position.

is a value that will add to the position in the z-axis to establish the stable grasping point. After the interpolation, the top grasp pose was applied to grasp the object directly. The first strategy in the actual environment was performed as shown in Figure 7((a)-(d)) and also as described in Algorithm 1 (lines 22 to 25).

• The Second Strategy.

The first strategy of the mobile manipulation was useful to grasp the objects in the back. However, we still are challenged to ensure stable grasping by the robot. For this reason, we developed a new strategy to grasp the objects in the back row to compensate for an inadequate object segmentation and robust and stable grasping. The objective of the **second strategy** was to grasp the objects from an 80cm high table while ensuring good stability. To pick up the objects, the algorithm for removing the objects in the front was conceived. The point of the strategy is that when the user selects the back row and target object, the robot calculates the centroid of the front row object as well. Then, the robot lifts the object off the front row and places it in the empty place on the table. First, to find the objects in the front, the function was implemented for searching the nearest distance between all objects and the target object. After the object in the front row was found, the pre-defined grasp position was applied. To ensure a stable grasping, the side grasp pose was introduced. Then, as the front row object was grasped (see Figure 8(a)), a pre-defined place was located at the right edge of the table (see Figure

Algorithm 1: The three mobile manipulation strategies.

Input : Joint position $\mathbf{q}$, Inital pose $\mathbf{x}_{init}$, Object row O_{row}, All objects information O_{all}, The centroid of the object selected and transformed O_{cen}, Object desired $\mathbf{x}_d$, Grasp pose $\mathbf{x}_{grasp}$, The new centroid of the object selected and transformed O_{newcen}, Distance of z axis $\mathbf{z}_{add}$, Table height T_h

Output: Goal pose, $\mathbf{x}_{goal}$

1 **while** *until manipulator finish task given* **do**
2 InitializationJacoArm($\mathbf{q}$);
3 TransformAllObjects;
4 $\mathbf{x}_{grasp} \leftarrow$ Pre-definedGraspPose($\mathbf{x}_{init}$, O_{row}, T_h);
5 **if** O_{row}.*back = True* **then**
6 **if** T_h <= *100* **then**
7 **• The Third Strategy Starts**
8 Update ObjectStates & MoveMobileBase;
9 $O_{newcen} \leftarrow$ Search ObjectUsingAxis(O_{cen});
10 $\mathbf{x}_d \leftarrow$ CalculateTargetObject(O_{newcen});
11 ManipulationBasedOnModeSelection;
12 $\mathbf{x}_{goal} \leftarrow \mathbf{x}_d \cdot \mathbf{x}_{grasp}$;
13 **end**
14 **if** T_h <= *87* **then**
15 **• The Second Strategy Starts**
16 Search NearestObject(O_{cen});
17 Search Pre-defiendEmptySpace(O_{all});
18 $\mathbf{x}_d \leftarrow$ Calculate TargetObject(O_{cen});
19 $\mathbf{x}_{goal} \leftarrow \mathbf{x}_d \cdot \mathbf{x}_{grasp}$;
20 **end**
21 **if** T_h <= *77* **then**
22 **• The First Strategy Starts**
23 Rotate MobileBase & Set the Angle of Neck
24 $\mathbf{z}_{add} \leftarrow$ InterpolationAddGraspAxis(O_{cen});
25 $\mathbf{x}_{goal} \leftarrow \mathbf{x}_{cen} \cdot \mathbf{x}_{grasp} \cdot \mathbf{z}_{add}$;
26 **end**
27 **end**
28 **end**

8(b)). Since the robot arm is mounted on the right of the body, we considered that an available place to the right would be easier. After the object was placed on the table, the arm returned to the initial position and the robot started grasping the target object with the side grasp pose (see Figure 8(c,d)). The entire process is represented in Algorithm 1 (lines 15 to 19).

• The Third Strategy.

The first and second strategies of the mobile manipulation were helpful to grasp objects in the back row, but the robot might fail to accomplish the task. For example, if the robot faced a table higher than its vi-

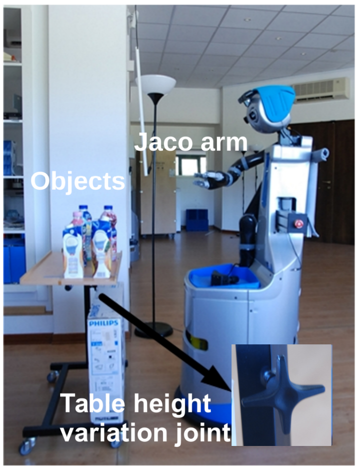

Figure 9: Experimental setup with multiple objects and an adjustable height table.

sual field, or if objects in the front row were taller than objects in the back row, the robot could not detect objects in the back row. For this reason, the **third strategy** was developed to grasp the hidden object. The third strategy is used in the particular situation of a hidden object when the first and second strategies cannot perform grasping tasks. In this strategy, human support was exploited to overcome the difficulty of object selection. To conduct a feasibility study for the third strategy, the hidden object was evaluated according to the decision of the user. The object selection method in the third strategy was not the same as in previous strategies because the user cannot see the object in the back using visual interface. However, the user already knows the location of the target object on the table and selects the back row and an object in the front using voice interface. After the user selects both row and object, the process of the third strategy, which is similar to that of the second strategy, is implemented. The basic difference between these two strategies is to update the state of the objects. After the object in the front is placed in the empty space, the robot should discover the target object. To find the object, the state of the scene was updated using the multi-object segmentation function. In addition, the information of the y-axis is used to find the target object because the target object is located colinear to the front object. The simplified algorithm is described in Algorithm 1(lines 7 to 11).

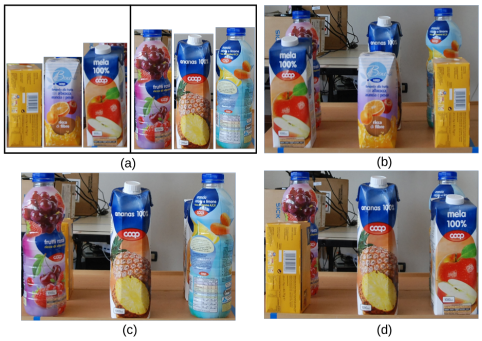

Figure 10: Objects were placed as follows: (a) left top: short objects, right: tall objects; (b) short objects in the front (FSO, right top); (c) tall objects in the front (FTO, left bottom); and (d) random size objects (RO, right bottom).

6 EXPERIMENTAL SETUP

Our experimental setup is shown in Figure 9. The robotic platform for the experiment is the Doro (domestic robot) personal robot (Cavallo et al., 2014), a service robot equipped with an omni-directional mobile base and a Kinova Jaco robotic arm. The Kinova Jaco arm, which has six degrees of freedom, is used for manipulation tasks. The head of the Doro is a pan-tilt platform equipped with two stereo cameras and an Asus Xtion Pro depth camera; they are used for object detection and segmentation. To implement ADL, we set up the experimental environment with multiple objects placed on the table, which can be adjusted in height as shown in Figure 9. Three objects were placed in the front, and the others were placed in the back. We used rectangular objects such as plastic bottles and juice boxes during the experiments.

For the experiment, several scenarios were organized. Before grasping an object in the back row, we tested a simple scenario for grasping an object in the front row. Then, the three manipulation strategies were tested to grasp an object in the back. The known table height was set at different steps of 10 cm, such as 70, 80, 90, and 100 cm. In addition, these three strategies were tested at an unknown table height to apply them in the real life situation. The objects were placed in three positions: short size objects in the front (FSO), tall size objects in the front (FTO), and random size objects (RO) (see Figure 10). During the manipulation with all strategies for unknown table height, we considered grasping one of the objects, which were placed randomly. The scenarios were evaluated 10 times for each strategy in terms of collision and success rate with known table height and with unknown table height.

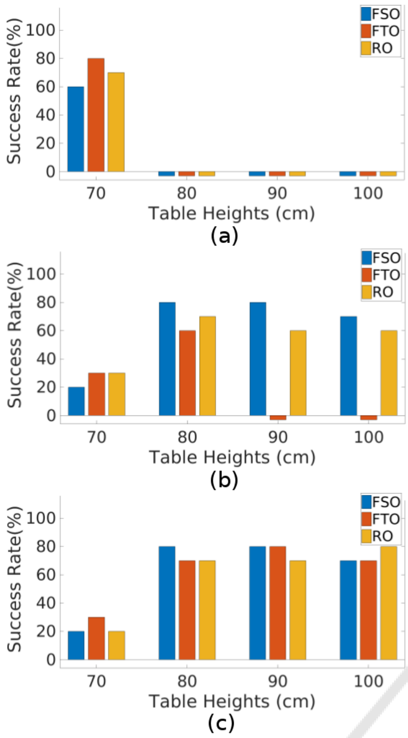

Figure 11: The success rate of the mobile manipulation for grasping an object in the back row on known table heights: (a) First strategy; (b) Second strategy; (c) Third strategy.

7 EXPERIMENTAL RESULTS

Firstly, quantitative analysis for three mobile manipulation strategies for known and unknown table height were performed.

7.1 Quantitative Analysis at Known Table Heights

The quantitative results focused on three criteria: success rates and collision The success rates were measured when the robot grasped a target object. We also considered a collision case in which objects were crashed into by a robot hand.

7.1.1 Success Rates

The success rates were evaluated in each strategy with three different object positions (FSO, FTO, and RO) on known table heights.

As shown in Figure 11(a), the success rate of the first strategy was higher for the 70 cm table height compared to any other table heights and strategies (60% to 80% for all three scenarios). At this table height, the second and third strategies have low success rates of manipulation because of lack of workspace. Nevertheless, some trials in the first strategy also failed to grasp the target object, although we developed linear interpolation to overcome insufficient object segmentation and stable grasping. In addition, except for the 70 cm table height, the first strategy was not successful in grasping because the robot cannot reach pre-grasp position over 77 cm.

The success rate of the second strategy was improved for a table height of greater than 80 cm, which is better than for the first strategy, but it doesn't work for the 70 cm table height (see Figure 11(b)). In particular, we found that the second strategy was a success for an average of 30% of the 70 cm table height tests in three different scenarios. Moreover, this strategy performed better for 80, 90, and 100 cm table heights for FSO and RO (success rate varies from 60% to 80%). We also observed that the second strategy failed to grasp FTO objects at table heights of 90 and 100 cm, which occurred due to taller objects blocking the target object (the human could not see or select the target object). In addition, when the robot grasped an object in the FTO, the robot only segmented small parts of objects in the back at 80 cm table height. Therefore, the grasp point was not extracted accurately.

Finally, the third strategy (see Figure 11(c)) could be carried out with any table height. In this case, the success rate varies from 70% to 80% (higher than second strategy) at 80, 90, and 100 cm table heights. However, for the 70 cm table height, the performance is similar to that of the second strategy (20% to 30%). As the robot removed the front object, the multi-object segmentation system was repeated automatically. As a result, the grasp point could be extracted more accurately than with the second strategy. Failure of the strategy occurred when the grasp force was insufficient to grasp the target object. Thus, the robot dropped the object during manipulation. As shown in Figure 11(c), the third strategy can be applied in any environment and shows better performance except for the 70 cm table height.

7.1.2 Collisions

During the evaluation, the number of collisions was measured for each table height using the three strategies for a total of 10 times for all scenarios in the experiments (see Figure 12).

The best results with 70 cm table height were achieved using the first strategy with a total average of seven collisions from all scenarios (see Figure 12(a)). The collisions in the strategy occurred while the robot arm returned to the home position. Except for the 70 cm table height, the low number of collisions oc-

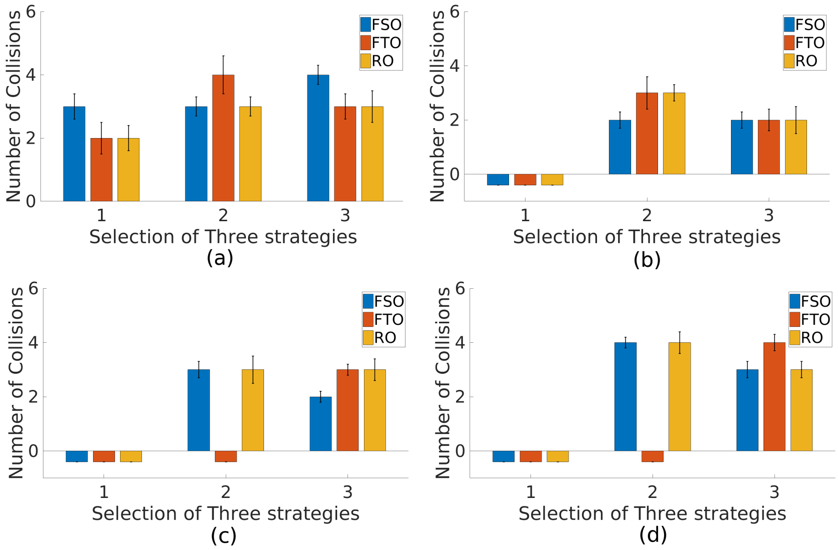

Figure 12: Number of collisions with three strategies during the grasp of an object in the back row for known table heights: (a) 70 cm table height; (b) 80 cm table height; (c) 90 cm table height; (d) 100 cm table height.

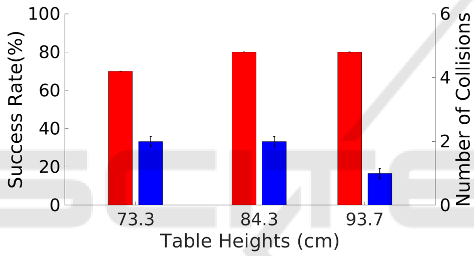

Figure 13: The success rate and number of collisions with three strategies of the mobile manipulation for grasping an object in the back row with unknown table heights.

curred at 80, 90, and 100 cm table height with the third strategy. The total number of collisions using the strategy occurred with all scenarios, with averages of six, eight, and ten for 80, 90, and 100 cm table height respectively (see Figure 12(b),(c),(d)), and standard deviation is about 5% of each collision. However, the second and third strategies have similar manipulations. Therefore, Figure 12(b),(c),(d) show that the collisions of the strategies are similar except for 90 cm and 100 cm table heights in the FTO scenario. The collisions with two strategies occurred while the robot arm was close to the object and returned to the home position with a target object.

Actually, with the first strategy, collisions only with the 70 cm table height could be measured because the robot arm could not reach objects with the other table heights (see Figure 12(a)). Moreover, we could not measure collisions with the second strategy in the FTO at the 90 and 100 cm table heights since the objects in the back were occluded due to being shorter than the front objects (see Figure 12(c),(d)).

7.2 Quantitative Results in Unknown Table Heights

Previous quantitative results were analyzed using known table height. However, various types of tables exist in reality. Before we set up the table height, we defined range between 70 and 100 cm to select strategies automatically. Then, the table height was set up randomly between defined ranges. Also, we only tested the strategies with objects in the RO configuration for implementing in the actual environment.

To confirm the three strategies of mobile manipulation, the three different table heights were measured and the results were evaluated in the same manner as previous cases (see Figure 13). The robot selected one strategy automatically to manipulate according to table height. As a result, the experiment was tested in ten trials; the average of the success rate of the manipulation in unknown table height is greater than 75%. We analyzed the number of collisions during the experiment. Collisions were evaluated with the same criteria, and an average of five collisions occurred with all three scenarios.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we present three mobile manipulation strategies in which the operator provides a simple command using visual and voice user interfaces. The three strategies of the mobile manipulation were developed to pick and place, and convey an object in the domestic environment effectively. Based on the results, the three strategies have their own advantages at the different table heights. Therefore, the intelligent strategy selection system can be applied for domestic environments that have different table heights.

Actually, the current system could be used to detect, cluster, and extract simple household objects such as bottles, boxes, etc. However, various objects that are different in shape exist in the domestic environment. Therefore, the 3D centroid of an object would not be able to grasp it. For this reason, we will develop a grasp pose algorithm for a variety of household objects with our strategies to save time (Redmon and Angelova, 2015). In addition, a deep learning-based approach for extracting grasping point could be considered to obtain more accurate performance (Lenz et al., 2015; Levine et al., 2016).

ACKNOWLEDGEMENT

The work described was supported by the Robot-Era and ACCRA project, respectively founded by the European Community's Seventh Framework Programme FP7-ICT-2011 under grant agreement No. 288899 and the Horizon 2020 Programme H2020-SCI-PM14-2016 under grant agreement No. 738251.

REFERENCES

Cavallo, F., Limosani, R., Manzi, A., Bonaccorsi, M., Esposito, R., Di Rocco, M., Pecora, F., Teti, G., Saffiotti, A., and Dario, P. (2014). Development of a socially believable multi-robot solution from town to home. *Cognitive Computation*, 6(4):954–967.

Chitta, S., Sucan, I., and Cousins, S. (2012). Moveit![ros topics]. *IEEE Robotics & Automation Magazine*, 19(1):18–19.

Ciocarlie, M., Hsiao, K., Jones, E. G., Chitta, S., Rusu, R. B., and Şucan, I. A. (2014). Towards reliable grasping and manipulation in household environments. In *Experimental Robotics*, pages 241–252. Springer.

Dogar, M. and Srinivasa, S. (2011). A framework for push-grasping in clutter. *Robotics: Science and systems VII*, 1.

Fromm, T. and Birk, A. (2016). Physics-based damage-aware manipulation strategy planning using scene dynamics anticipation. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 915–922. IEEE.

Hershberger, D., Gossow, D., and Faust, J. (2011). Rviz.

Kitaev, N., Mordatch, I., Patil, S., and Abbeel, P. (2015). Physics-based trajectory optimization for grasping in cluttered environments. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3102–3109. IEEE.

Leeper, A. E., Hsiao, K., Ciocarlie, M., Takayama, L., and Gossow, D. (2012). Strategies for human-in-the-loop robotic grasping. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 1–8. ACM.

Lenz, I., Lee, H., and Saxena, A. (2015). Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. (2016). End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40.

Ozgür, A. and Akın, H. L. Planning for tabletop clutter using affordable 5-dof manipulator.

Parasuraman, R., Sheridan, T. B., and Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans*, 30(3):286–297.

Pitzer, B., Styer, M., Bersch, C., DuHadway, C., and Becker, J. (2011). Towards perceptual shared autonomy for robotic mobile manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 6245–6251. IEEE.

Redmon, J. and Angelova, A. (2015). Real-time grasp detection using convolutional neural networks. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1316–1322. IEEE.

Rusu, R. B. and Cousins, S. (2011). 3d is here: Point cloud library (pcl). In *Robotics and automation (ICRA), 2011 IEEE International Conference on*, pages 1–4. IEEE.

Sak, H., Senior, A., Rao, K., Beaufays, F., and Schalkwyk, J. (2015). Google voice search: faster and more accurate. *Google Research blog*.

Schraft, R., Schaeffer, C., and May, T. (1998). Care-o-bot/sup tm: the concept of a system for assisting elderly or disabled persons in home environments. In *Industrial Electronics Society, 1998. IECON'98. Proceedings of the 24th Annual Conference of the IEEE*, volume 4, pages 2476–2481. IEEE.

Stilman, M., Schamburek, J.-U., Kuffner, J., and Asfour, T. (2007). Manipulation planning among movable obstacles. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3327–3332. IEEE.

Torta, E., Oberzaucher, J., Werner, F., Cuijpers, R. H., and Juola, J. F. (2012). Attitudes towards socially assistive robots in intelligent homes: results from laboratory studies and field trials. *Journal of Human-Robot Interaction*, 1(2):76–99.

Trevor, A. J., Gedikli, S., Rusu, R. B., and Christensen, H. I. (2013). Efficient organized point cloud segmentation with connected components. *Semantic Perception Mapping and Exploration (SPME)*.