The Helping Hand: An Assistive Manipulation Framework Using Augmented Reality and a Tongue-Drive

Fu-Jen Chu, Ruinian Xu, Zhenxuan Zhang, Patricio A. Vela, and Maysam Ghovanloo

Abstract—A human-in-the-loop system is proposed to enable collaborative manipulation tasks for persons with physical disabilities. Studies have shown the cognitive burden of subject is reduced with increased autonomy of assistive system. Our framework is aim to communicate high-level intent from subject and generate desired manipulations. We elaborate a framework by incorporating a tongue-drive system and a 7 DoF robotic arm through an virtual interface. The assistive system processes sensor input for interpreting user environment, and the subject provides an ego-centric visual feedback via interface to guide action loop for achieving the tasks. Extensive experiments are performed on our framework and we show our coupled feedback loops is able to effectively simplify complex manipulation tasks.

I. INTRODUCTION

Paralysis afflicts 5.5 million people in the United States [?]. Persons with high-level paralysis (both upper and lower extremities) rely on the care of others and the modification of their environment for accomplishing the activities of daily living (ADL). While the introduction of technology by way of personal mobility devices, (such as wheelchairs) and electronic environmental control systems provides some level of autonomy [1], [2], there is still a gap between the activities afforded by these interventions and the full needs of the paralyzed population regarding the ADL. Research and translational efforts in robotic and computational assistive technologies (AT) indicate that these emerging intervention and support technologies can bridge the existing ability gap. However, effective interface and command modalities plus sufficiently intelligent robotic and computational technologies are needed for AT to be of practical utility and have wide adoption [3].

Assistive robotic manipulators have long been considered as enabling technologies for self-supportiveness and independence in accomplishing activities of daily life (ADLS) for individuals who have upper extremity disabilities [4], [5]. Commonly seen assistive robotic arms such as the JACO arm and the MANUS have 6-7 degrees of freedom, and admit execution of many ADL [6], [7]. However, it is often challenging for people with paralysis of arms to fully control an assistive system at the required proficiency level [8]. Even for non-paralyzed populations, the traditional manipulator control interfaces, such as the ring-and-arrow marker to control translation and rotation, as well as open/close the end-effector, require some level of expertise. Directly moving an end-effector or setting an end pose with markers for autonomous path planning often involves human operation error [9]. Better performance can be achieved by increasing robot autonomy [10]. Hence there is a need to develop effective interfaces to effectively communicate human intent to a manipulator for completing tasks.

In order to communicate high level human intent to a robot arm, the user should ideally interact with an easily accessible interface. Brain, head, and intraoral devices have been studied as suggested approaches [11]–[13]. Among those, tongue-drive system (TDS) shows promise to be a versatile interface due to both its aesthetics and to the high selectivity of human tongue. The eTDS is a wireless assistive technology that can translate the tongue motion to various commands [13]. It was first designed for individuals with severe disabilities to control their environments [14], [15].

To simplify complex manipulation and enhance autonomy of assistive system, human intent can be summarized on a virtual canvas. Possible display candidates for menuing systems include laptops, tablets, audio assistants, and augmented reality glasses [16]. Augmented reality glasses provides a virtual menu which is able to reflect the interpretation of assistive system and interact with user feedback via TDS by selecting corresponding intent on a virtual canvas.

In this work, we study the feasibility of incorporating TDS with augmented reality to leverage the user’s egocentric perspective for enabling autonomy in an assistive system. The user will provide cues to the assistive robotic manipulator for task execution. The main contributions of this paper are:

(1) A streamlined framework to enable human-robot collaborative manipulation tasks. Our design shows the feasibility to utilize both a virtual interface and an intraoral tongue-drive to couple high-level user command feedback and the low-level robotic components for manipulation tasks;

(2) An architecture designed to integrate augmented reality glasses with the tongue-drive in the loop of a robotic arm planner. The human-robot interface performs object recognition, localization, and planning to minimize the cognitive burden of the user when completing manipulation tasks; and

(3) Experiments are performed on manipulation tasks. We show the functionality of each component and illustrate the effectiveness of the collaborative human-robot framework regarding hands-free manipulator control.

II. RELATED WORK

Assistive robots for persons with disabilities increase user independence and enhance physical functionality. They can be broadly categorized into companion robots apart
from wheelchairs, and robotic devices integrated with wheelchairs, for which the wheelchair mounted robotic arm (WMRA) is one such example [7]. The former category consists of static manipulators and mobile manipulators. Static manipulators are typically installed or placed where needed and cannot move on their own. Mobile manipulators can aid a person with different tasks that share common manipulation strategies [4], [5], [10], [16]. The work described here fits in the static manipulator category.

Early assistive manipulation research focused on using existing interfaces (joystick, touch-screen) to directly control the robot arms, much like most commercial products today. Contemporary work focuses on supervisory or shared control with efforts going towards designing interfaces for facilitating and simplifying the user experience improving the design for performance and improving perception, planning, and control algorithms for increased autonomy [17]–[22].

work in This paper describes how to leverage contemporary advances in computer vision and robotic planning to increase the autonomy of assistive manipulation robots, while using new tongue-drive technology for hands-free operation.

A key component of the assistive robotic system is the use of augmented reality (AR) glasses as the interface for communicating high-level human intents. Augmented reality glasses, first designed for gaming, allow users to visualize virtual objects in real environments. With augmented reality glasses a user can control a virtual menu or program [23]. Recently AR has been applied to the rehabilitation and assistive systems fields. Explored use cases include education for cognitively disabled elementary school children [24], and for surgical robotics [25] which described a framework for human robot collaboration, whereby hand gestures are used for communication with the robot. Additionally, [26] develops a human-robot collaborative mechanism involving a hand prosthesis to assess the grasping ability of the user to provide grasping feedback via AR.

A recently developed system whose overall design is quite similar to the one described here [27] uses eye gaze and EEG as the two input modalities. The user can perform pick and place activities with a low DOF robotic arm by fixating their gaze and activating the EEG via thought. Intermediate phases of the routine must be controlled by the user through the assistance of a nearby monitor that provides augmented reality feedback. The AR approach was shown to improve task performance (time and error) relative to the lack of AR. Our work seeks to develop a more autonomous robotic system through the integration of modern computer vision algorithms and advanced robotic planning methods. Our input modalities are a headworn AR system (through head fixation) and a tongue drive system. The feedback loop coupling is done through the head-mounted AR system.

III. System Architecture

This section describes in detail the human-robot collaborative system, after first introducing an overview of the human-in-the-loop framework.

Figure 1 shows the overview of the building blocks. In our design, the input to the robot component received from the augmented reality (AR) sensors consists of RGB-D image sequences. The visual signals are passed to the vision system. After performing processing and interpretation of the scene, it generates a corresponding virtual menu. The menu is populated with actions that the robot arm may perform on behalf of the user. Once the AR presents the virtual menu to the user, it waits for the user’s intent as feedback. Due to the need for hands-free operation, a tongue-drive system (TDS) is the interface for human input. The TDS supports button press operations for navigating the virtual menu. The selected intent received by AR will then trigger the manipulator to autonomously complete complex tasks.

In what follows, we describe in detail the vision system (§III-A), the augmented reality sensor system (§III-B), the TDS (§III-C), and the manipulator (§III-D).

A. Vision Interpretation of Users Environment

1) object detection: To effectively interact with objects, it is important to recognize them for the purpose of establishing what manipulation behaviors can be applied to objects in the scene. We build our main vision system based on a deep model for interpreting target objects to be manipulated. The vision system adopts the state-of-the-art
YOLO [28] with twenty-four convolutional layers followed by 2 fully connected layers. Compared to existing state-of-the-art detection methods [29]–[33] including two-staged architectures such as Fast-RCNN [34] and Faster-RCNN [35], YOLO elaborates a simpler pipeline as an unified architecture which is potentially fast in speed. Fast versions of YOLO achieve more than 150 fps, which is suitable for vision-based applications requiring real-time processing.

YOLO models the detection as regression problem, which divides an input image into an $S \times S$ grid. Each grid cell predict $B$ bounding boxes, $B$ confidences and $C$ class probabilities. Each bounding box consists of 4 parameters: $x, y, w, h$. We followed the same training procedure and optimize with multi-part loss function as follow:

\[
\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^b [(x_i + \hat{x}_i)^2 + (y_i + \hat{y}_i)^2]
\]

\[
+ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^c [(\sqrt{w_i} + \hat{w}_i)^2 + (\sqrt{h_i} + \hat{h}_i)^2]
\]

\[
+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^d (C_i + \hat{C}_i)^2
\]

\[
+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^e (C_i + \hat{C}_i)^2
\]

\[
+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \sum_{C \in \text{classes}} (p_i(c) + \hat{p}_i(c))^2
\]

where the $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i$ denotes the ground truth bounding box configurations. $1^b_{ij}$ denotes that cell $i$ contains object. And $1^d_{ij}$ denotes that jth bounding box predictor in cell $i$. In this paper we have $S = 7, B = 2$, and we use $C = 20$ for pretrain, and $C = 3$ for finetuning. A detailed architecture of YOLO is shown in Fig 2. Interested reader is encouraged to refer to [28] for designed and implementation details.

To operate with high accuracy with regards to our intended application, we begin with a YOLO model pretrained on the PASCAL VOC 2007 train/val + 2012 train/val datasets. Fine-tuning uses our manually collected dataset of office table objects. Our batchsize is set to 64, and learning rate is set to 10⁻³ and decreases by 10 every 10 epochs.

2) object localization: The object location with respect to the manipulator is required for assistive system to interact with targets. To simplify the overall system, one assumption made in our scenario is that the manipulator base is fixed. We employ an ARUCO marker in the field of view and presume it to be visible when the user looks at objects of interest. Upon seeing the ARUCO marker, the system registers the camera frame relative to the manipulator base frame. The 2D bounding box output from the object detection stage is processed against the calibrated depth image to cropped the point cloud region of interest of the object for added processing. Applying a region growing segmentation algorithm [37] on the point cloud within the region of interest, the largest cluster is kept as a denoising step. After removing the points belonging to the table surface, objects of interest are extracted so that their 3D bounding boxes can be obtained for object localization and manipulation purposes. Figure 2 visualizes the pipeline for going from the object bounding box in the image plane to the actual 3D bounding box in the reconstructed object point cloud in the manipulator base frame.

3) grasping orientation: Graspable locations for robotic manipulation are determined by a second deep model. We employed the 5D grasp rectangle representation for a grasping location, which is a simplification of Jaing et al.’s 7D representation [38]. The 5D representation parametrizes a grasp candidate by the location and orientation of a parallel plate gripper, plus feasible gripper dimensions, $g = \{x, y, w, h, \theta\}$. The coordinates $(x, y)$ are the center of the rectangle, $\theta$ is the orientation of the rectangle relative to the horizontal axis, $w$ is the width and $h$ is the height.

Due to finer bounding boxes is required for successful manipulation, we adopt ResNet-50 with fifty layers and employ it into two-stage object detection architecture similar to RCNN. Our modified model is shown in Fig 3. By transforming grasp orientation from regression to classification problem, we define the total loss to be the sum of grasp proposal net loss $L_{gpn}$ and grasp configuration regression net loss $L_{gcr}$:

\[
L_{\text{total}} = L_{gpn} + L_{gcr}
\]

where $g_t$ denote the 4-dimensional vector specifying the reset $(x, y, w, h)$ of the $i$-th grasp configuration, and $g_i$ denote the probability of the $i$-th grasp proposal, we define the loss of grasp proposal net to be:

\[
L_{gpn}(\{(g_i, g_t^i)\}_{i=1}^{C}) = \sum_{i} L_{\text{cls}}(g_i, g_t^i) + \lambda \sum_{i} g_i^* L_{\text{reg}}(g_t, g_t^*)
\]

where $g_t^* = 0$ if no grasp and $g_t^* = 1$ if grasp is specified. $t^*$ is the ground truth grasp coordinates corresponding to $g_t^*$.

Let $p_c$ denote the probability of class $c$ after softmax layer, and $t_c$ denote the corresponding predicted grasp bounding box, we define the loss of grasp configuration prediction by

\[
L_{gcr}(\{(p_c, t_c)\}_{c=0}^{C}) = \sum_{c} L_{\text{cls}}(p_c) + \lambda_2 \sum_{c} 1_{c \neq 0} L_{\text{reg}}(t_c, t^*_c)
\]

where $t^*_c$ is the ground truth grasp bounding box.

To avoid over-fitting and achieve better minimum point, we share the pretrained weights of ResNet-50 on COCO-2014 and train on Cornell dataset. We train our model for 5 epochs with learning rate starting at 0.0001 and divide by 10 every 10000 iterations. For experiment in this paper, we select the grasp candidate which is approachable from top-down view with highest confidence score and apply the predicted orientation for execution.
B. Interface: Egocentric Vision and User Intent

Vision algorithm simplifies the complex manipulation tasks by identifying object class, localizing object pose and predicting grasping candidates. To bridge the vision assistive system and user intent, we propose to incorporate a virtual interface with augmented reality glasses for effective and efficient interaction. Compared to interfaces such as mobile phone and audio assistive system, providing visual feedback to the user in their field of view does not require gaze to be broken from the object of interest. Using augmented reality to visualize actions and provide context-based menuing systems tends to be more efficient and well-understood [39].

Users intent can be communicated as feedback in the human-robot loop through fixation and tongue drive input. Since the AR system has visual sensors that provide the ego-centric view to the controlling computer of the robotic arm, the robot has a similar visual perspective a the user. Visually

A META-1 glasses is adopted in our collaborative framework. META glasses are equipped with a color camera, a time-of-flight depth sensor, and an IMU. Its total weight is 0.3 kilograms. The AR projected field of view is 23 degrees with 960x540 resolution for each eye. For development, META supports Windows 32bit/64bit platform and Unity3D engine with Unity SDK for META.

As shown in Fig. 3, we designed corresponding virtual interfaces for detected objects. The interface serves as menu on Unity3D canvas with interactive buttons, which can be directly controlled by tongue-drive system. In our framework, the users environment is first processed by vision system, and a corresponding virtual interface is shown and waiting for the high-level feedback from users. According to user’s intent, assistive system is able to complete complex manipulation tasks.

C. Interface: The Intraoral Tongue-Drive

The TDS headset contains five magnetic sensors (2 near each side of the cheek, 1 on the top of the head) held by the custom 3D printed components and a commercially available headgear. These sensors are used to locate the position of the disk magnet (D21B-N52, K&J Magnetics, Inc.) that is temporarily attached to the tip of the tongue using tissue adhesives. Communication with the computer is through Bluetooth Low Energy. Prior to use, the TDS requires a training stage to remove the effect of the external magnetic field (EMF). During EMF cancelation calibration, 1000 magnetic sensor data are collected while the user keeps their tongue in a resting position and rotates their head. Since the magnetic sensor on the top of the head is only influenced by EMF, the EMF portion of the other 4 sensors can be cancelled using linear least square fitting and coordinate transformation. A follow-up training period asks the user to provide training data for each command (up, down, left, right, and rest) 3 times in randomized order. Then, an RBF SVM model is trained. During operation, the SVM based algorithm performs classification using 10 past EMF cancelled data every 10 ms. If all the classification results are the same, the command is sent to the robotic arm.

D. Interface: Human Intent and Autonomous Manipulation

Through the augmented reality interface, high level human intent may be selected and triggered by the tongue-drive system. A robotic arm designed for autonomous manipulation, shown in Fig. 4, is a 7 DOF manipulator. The extra DOF enables end-effector orientation constraints during manipulation. The manipulator model was created in SolidWorks and ported to ROS for kinematics. The gripper, as an end-effector, is designed for general grasping purpose yet can be easily replaced with different ends.

Path planning for manipulation is performed via a modified MoveIt! package in ROS. The modification admits path planning with mixed initial and final configurations [40], [41], the former are the joint angles and the latter is an end-effector configuration, thereby avoiding the need for
inverse kinematics. The grasping task relies on the object location and the approaching direction as estimated by the vision system; they get input to the path planner model of manipulator. Once a human intent is triggered, the manipulator autonomously completes the tasks without detailed interactions such as Cartesian movement or specifying grasp poses.

With all the components integrated, we close the human-robot collaborative framework loop. By increasing the autonomy of assistive system, we aim to lower their cognitive burden to the user, speed up the processing duration, and decrease the failure rate due to human error. Our experiments, which follow in the next section, are also an test of the autonomous capabilities of the robotic arm. As the underlying algorithms improve, the entire closed-loop system will improve.

IV. EXPERIMENTS AND EVALUATION

Three experiments were performed to test the overall system. The first tests out the manipulator error, given that it is constructed from off-the-shelf components. The other two test out the human-robot closed-loop manipulation system by performing pick and place tests for two different objects, a cup and a screwdriver.

A. Object Location Error

In this experiment, we tested the accuracy of target 3D bounding box estimation. We are interested to know the how the error is accumulated by using the the point cloud and 2D bounding box output. A cup was first trained to be recognized by YOLO using the finetuned approach description earlier. We compared the predicted the upper center with the ground truth. In the experiment settings, we randomly select 6 location within the reachable area and within the field of view of META. For each location, we repeat 3 trials. We then computed the mean error and variance for all 18 trials. Table I provides the results of the object location accuracy test. Localization accuracy is sufficiently small that the bounding box will accurately bound most objects of interest, i.e., those that would actually be graspable by the robotic arm.

| axis | x | y | z(height) |
|------|---|---|-----------|
| mean | 0.974 | 1.094 | 1.469 |
| var  | 0.294 | 0.141 | 0.125 |

B. End-effector Location Error

Testing the error of the manipulator involved measuring the difference between the actual frame center of the link which is connected to the end-effector and the specified final location. We chose this link since, in our design, the last joint is made for controlling the open/close of the end-effector. During motion planning, we approach the target by planning this link to move to the targeted position. Table II has the outcome of the end-effector accuracy test. Though not as precise as more expensive, commercial robotic arms, the precision is small enough that visual servoing should sufficiently correct the small error.

| axis     | x     | y     | z(height) |
|----------|-------|-------|-----------|
| mean     | 0.294 | 1.528 | 1.315     |
| var      | 0.049 | 0.293 | 0.129     |

C. Pick and Place with a Cup

In this section we combine the 3D bbs with manipulator and test on pick and place. In this experiment, the entire system is in operation, waiting for the user to select the proper action. The goal is to pick up a light cup from the top and place on user specified location on the table, once commanded to by the user. The RGBD image is first processed by object detector and 3D bbs for location to pick up (triggered by tongue drive on the user menu shown on AR). A marker is shown on the center of the image for user to specify the location for placing. Success for placing means we place to the target location within a 1cm larger boundary. Table III provides the outcomes of the 10 test trials. In one case, the robotic arm was not able to pick up the cup due to incorrect positioning.

D. Pick and Place with a Screwdriver

Here we perform a similar pick and place experiment, but with a different object: a screwdriver. The goal is for the user to pick up the screwdriver from the table and place it at a user specified location on the table. The RGBD image is first processed by object detector and 3D bbs for location as well as the orientation to initiate the pick motion (triggered by tongue drive on the user menu shown on AR). The procedure works as described in the first pick and place experiment. Success for placing means we place to the target location within 1cm larger boundary. Table IV provides the timing and success information for the task. In two cases, the robotic arm was not able to pick up the screwdriver, and in three cases where it did pick up the screwdriver, it fell from the gripper. The main source of error was due to the center of mass being off relative to the visual centroid of the screwdriver. Most of the failure cases are due to losing grip of the screwdriver.

V. CONCLUSION

We present a collaborative framework for human with disabilities to perform desired manipulations. Our proposed assistive system enhances autonomy by integrating vision algorithms with augmented reality and a intraoral tongue-drive device. The human-in-the-loop framework communicates intent and achieves tasks by effectively simplifying
the complexity of controlling manipulation task. We perform experiments to illustrate the effectiveness of our system and analyze the speed and success rate. Future studies will include experimental studies with human subject test and analyze the cognitive burden, as well as explore the addition of visual servoing algorithms to enhance manipulation performance.

### REFERENCES

[1] D. Ding, R. Cooper, B. Kamiński, J. Kanaly, A. Allegretti, E. Chaves, and S. Hubbard, “Integrated control and related technology of assistive devices,” Assistive Technology, vol. 15, no. 2, pp. 89–97, 2003.

[2] B. Yousefi, X. Huo, J. Kim, E. Veledar, and M. Ghovanloo, “Qualitative and quantitative assessment of learning in a tongue-operated computer input device-part ii: Navigation tasks,” IEEE Transactions on Information Technology in Biomedicine, vol. 16, no. 4, pp. 633–643, 2012.

[3] R. Wong, R. Hellman, and V. Santos, “Spatial asymmetry in tactile sensor skin deformation aids perception of edge orientation during haptic exploration,” IEEE Transactions on Haptics, vol. 7, no. 2, pp. 191–202, 2014.

[4] B. Graf, M. Hans, and R. Schraft, “Care-o-bot II - development of a next generation robotic home assistant,” Autonomous Robots, vol. 16, no. 2, pp. 193–205, 2004.

[5] P. Deegan, R. Grupen, A. Hanson, E. Horrell, S. Ou, E. Riseman, S. Sen, B. Thibodeau, A. Williams, and D. Xie, “Mobile manipulators for assisted living in residential settings,” Autonomous Robots, vol. 24, no. 2, pp. 179–192, 2008.

[6] Kinova, “Jaco rehab edition.” 2014. [Online]. Available: \url{http://kinovarobotics.com/products/jaco-rehab-edition/}

[7] D. Kim, Z. Wang, and A. Behal, “Motion segmentation and control design for UCF-MANUS - An intelligent assistive robotic manipulator,” IEEE-ASME Transactions on Mechatronics, vol. 17, no. 5, pp. 936–948, 2012.

[8] L. A. Struijk and R. Loutis, “Comparison of tongue interface with keyboard for control of an assistive robotic arm,” in IEEE International Conference on Rehabilitation Robotics, 2017, pp. 925–928.

[9] S. C. D. Kent, C. Saldanha, “A comparison of remote robot teleoperation for general object manipulation,” in ACM/IEEE International Conference on Human-Robot Interaction, 2017, pp. 371–379.

[10] T. Chen, M. Ciocarlie, S. Cousins, P. Grice, K. Hawkins, K. Hsiao, C. Kemp, C. King, D. Lazewatsky, A. Leeper, H. Nguyen, A. Paepecke, C. Pantofaru, W. Smart, and L. Takayama, “Robots for humanity: Using assistive robotics to empower people with disabilities,” IEEE Robotics and Automation Magazine, vol. 20, no. 1, pp. 30–39, 2013.

[11] K. Lyons and S. Joshi, “Paralyzed subject controls telepresence mobile robot using novel sEMG brain-computer interface: Case study,” in IEEE International Conference on Rehabilitation Robotics, 2013.

[12] C. Martins Pereira, R. Bolliger Neto, A. Reynolds, M. de Miranda Luzo, and R. Oliveira, “Development and evaluation of a head-controlled human-computer interface with mouse-like functions for physically disabled users,” Clinics, vol. 64, no. 10, pp. 975–981, 2009.

[13] X. Huo, J. Wang, and M. Ghovanloo, “A magneto-inductive sensor based wireless tongue-computer interface,” IEEE transactions on neural systems and rehabilitation engineering, vol. 16, no. 5, pp. 497–504, 2008.

[14] X. Huo and M. Ghovanloo, “Evaluation of a wireless wearable tongue–computer interface by individuals with high-level spinal cord injuries,” Journal of neural engineering, vol. 7, no. 2, p. 026008, 2010.

[15] J. Kim, H. Park, J. Bruce, E. Sutton, D. Rowles, D. Pucci, J. Holbrook, J. Minocha, B. Nardone, D. West, et al., “The tongue enables computer and wheelchair control for people with spinal cord injury,” Science translational medicine, vol. 5, no. 213, pp. 213ra166–213ra166, 2013.

[16] “Evaluation of a graphic interface to control a robotic grasping arm: A multicenter study,” Archives of Physical Medicine and Rehabilitation, vol. 90, no. 10, pp. 1740–1748, 2009.

[17] C. Dune, C. Leroux, and E. Marchand, “Intuitive human interaction with an arm robot for severely handicapped people - a one click approach,” in IEEE International Conference on Rehabilitation Robotics, 2007, pp. 582–589.

[18] A. EDSINGER and C. KEMP, “Human-robot interaction for cooperative manipulation: Handing objects to one another,” in IEEE International Symposium on Robot and Human Interactive Communication, 2007.

[19] P. Grice and C. Kemp, “Assistive mobile manipulation: Designing for

---

**TABLE III**

| trial | time for pickup after trigger(s) | success? | time for place after trigger(s) | success? |
|-------|---------------------------------|----------|---------------------------------|----------|
| 01    | 22.47                           | Y        | 53.12                           | Y        |
| 02    | 22.39                           | Y        | 53.86                           | Y        |
| 03    | 21.77                           | Y        | 52.30                           | Y        |
| 04    | 22.79                           | Y        | 51.70                           | Y        |
| 05    | 21.88                           | Y        | 51.04                           | Y        |
| 06    | 21.71                           | Y        | 51.11                           | Y        |
| 07    | 22.25                           | Y        | 51.14                           | Y        |
| 08    | n/a                             | N        | n/a                             | N        |
| 09    | 22.38                           | Y        | 51.34                           | Y        |
| 10    | 21.91                           | Y        | 51.40                           | Y        |

**TABLE IV**

| trial | time for pickup after trigger(s) | success? | time for place after trigger(s) | success? |
|-------|---------------------------------|----------|---------------------------------|----------|
| 01    | 27.41                           | Y        | 58.50                           | Y        |
| 02    | 27.49                           | Y        | 59.07                           | Y        |
| 03    | 27.25                           | Y        | n/a                             | N (fall down) |
| 04    | 27.50                           | Y        | n/a                             | N (fall down) |
| 05    | 27.78                           | Y        | 60.39                           | Y        |
| 06    | 27.25                           | Y        | n/a                             | N (fall down) |
| 07    | 27.28                           | Y        | 58.11                           | Y        |
| 08    | n/a                             | N        | n/a                             | N        |
| 09    | 27.39                           | Y        | 58.04                           | Y        |

---
operators with motor impairments,” in RSS Workshop on Socially and Physically Assistive Robotics for Humanity, 2016.

[20] D. Kim, R. Hazlett-Knudsen, H. Culver-Godfrey, G. Rucks, T. Cunningham, D. Portee, J. Bricout, Z. Wang, and A. Behal, “How autonomy impacts performance and satisfaction: Results from a study with spinal cord injured subjects using an assistive robot,” IEEE Transactions on Systems Man and Cybernetics Part A- Systems and Humans, vol. 42, no. 1, pp. 2–14, 2012.

[21] D. Kim, Z. Wang, N. Paperno, and A. Behal, “System design and implementation of UCF-MANUS - an intelligent assistive robotic manipulator,” IEEE-ASME Transactions on Mechatronics, vol. 19, no. 1, pp. 225–237, 2014.

[22] C. Chung, H. Wang, and R. Cooper, “Functional assessment and performance evaluation for assistive robotic manipulators: Literature review,” Journal of Spinal Cord Medicine, vol. 36, no. 4, pp. 273–289, 2013.

[23] R. T. Azuma, “A survey of augmented reality,” Presence: Teleoperators and virtual environments, vol. 6, no. 4, pp. 355–385, 1997.

[24] E. Richard, V. Billaudou, P. Richard, and G. Gaudin, “Augmented reality for rehabilitation of cognitive disabled children: A preliminary study,” in Virtual Rehabilitation, 2007. IEEE, 2007, pp. 102–108.

[25] R. Wen, W.-L. Tay, B. P. Nguyen, C.-B. Chng, and C.-K. Chui, “Hand gesture guided robot-assisted surgery based on a direct augmented reality interface,” Computer methods and programs in biomedicine, vol. 116, no. 2, pp. 68–80, 2014.

[26] M. Markovic, S. Dosen, C. Cipriani, D. Popovic, and D. Farina, “Stereovision and augmented reality for closed-loop control of grasping in hand prostheses,” Journal of neural engineering, vol. 11, no. 4, p. 046001, 2014.

[27] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in Advances in neural information processing systems, 2015, pp. 91–99.

[28] R. Munoz-Salinas, “Aruco: a minimal library for augmented reality applications based on opencv,” Universidad de Cordoba, 2012.

[29] “Point cloud library.” [Online]. Available: \url{http://www.pointclouds.com}