

# Three Dimensional Computer Vision-Based Alternative Control Method for Assistive Robotic Manipulator

Hyun W. Ka, PhD; Dan Ding, PhD; Rory A. Cooper, PhD

Human Engineering Research Laboratories, Department of Veterans Affairs, Pittsburgh, PA  
Department of Rehabilitation Science and Technology, University of Pittsburgh, Pittsburgh, PA  
6425 Penn Ave. Suite 400 Pittsburgh PA 15206

## Abstract

JACO (Kinova Technology, Montreal, QC, Canada) is an assistive robotic manipulator that is gaining popularity for its ability to assist individuals with physical impairments in activities of daily living. To accommodate a wider range of user population especially those with severe physical limitations, alternative control methods need to be developed. In this paper, we presented a vision-based assistive robotic manipulation assistance algorithm (AROMA) for JACO, which uses a low-cost 3D depth sensing camera and an improved inverse kinematic algorithm to enable semi-autonomous or autonomous operation of the JACO. The benchtop tests on a series of grasping tasks showed that the AROMA was able to reliably determine target gripper poses. The success rates for the grasping tasks ranged from 85% to 100% for different objects.

*Keywords: rehabilitation robotics, vision-based robot control, alternative robotic manipulation, human-robot interaction, assistive technology*

## I. INTRODUCTION

Assistive robotic manipulators have long been recognized to have great potential to assist individuals with physical disabilities in a range of activities of daily living (ADLs) [1-6]. In 2013, the U.S. Department of Veteran Affairs prescribed about 170 assistive robotic manipulators to veterans with disabilities to support their independent living. One of the most popular assistive robotic manipulators is JACO from Kinova Technology (Montreal, QC, Canada). JACO is composed of six inter-linked segments with a three-fingered hand. A default control method for JACO is a 3D joystick with 7 buttons and a knob. However, people who have severely impaired motor functions or have a combination of multiple disabilities have found it

difficult or impossible to independently operate it [3].

Vision-based autonomous control has been investigated as one of the solutions to accommodate people who cannot effectively use the manual control methods [7-15]. Vision-based control can transfer the loading in positioning and fine manipulation to the autonomous algorithm to reduce the complexity exposed to the user. To implement the vision-based autonomous control, many researchers adopted an eye-in-hand camera [7, 8, 13, 15], on the robot gripper or wrist to guide the robot towards an object of interest. This approach needs to update the object locations continuously until the end-effector acquired the target object, and thus is computationally expensive. Some researchers mounted a camera on the fixed position at the robot base or shoulder [12]. While this approach has advantage of finding a path and grasping plan even when the object is occluded from the starting location or folding position [14], it requires the knowledge of the target object as well as the surroundings in advance to localize the target object and plan a trajectory [16]. Other researchers use the combination of the above two approaches to provide more reliable and robust control [10, 14]. However, the combined approach can significantly increase the implementation cost and system overhead. More recently, the use of 3D camera like Microsoft Kinect has been investigated in assistive robot applications [11].

We have developed and evaluated a vision-based assistive robotic manipulation assistance algorithm (AROMA) for JACO, which uses another kind of low-cost 3D depth sensing camera and an improved inverse kinematic (IK) algorithm over the IK algorithm provided by the JACO Application Program Interface (API). In addition, AROMA was developed on a Windows operating system instead of the Robot Operating System (ROS), which makes it easier for the algorithm to be adopted by non-technical users and clinical professionals.

## II. DESIGN APPROACH

AROMA consists of two inter-related modules: 3D vision module and custom IK module. The following diagram (Figure 1) shows how the AROMA works.

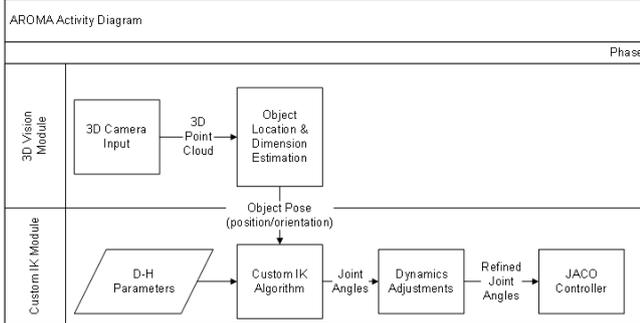


Figure 1. AROMA Activity Diagram

### A. 3D Vision module

We adopted a low-cost short-range 3D depth-sensing camera (Senz3D manufactured by Creative Labs, Inc., Milpitas, CA) in this study and mounted it on the robot base (Figure 2). The Senz3D uses the 'time-of-flight' technique to obtain depth information within its field of view (diagonal 70 degrees) and working range (20-90m) at a maximum resolution of 320x240. It is known that the 'time-of-flight' technique in general outperforms the structured light technique used in Microsoft Kinect [17]. The Senz3D can generate a 3D point cloud in which each point represents the distance (15-90cm) to objects within its field of view. Based on the 3D point cloud, the shape and the dimensions (width and height) of the target object can be estimated. To stabilize the depth data and increase the accuracy of the estimation, we used depth processing techniques provided by RGBD module (by Vincent Ronald) integrated into Intel's open source computer vision library (OpenCV 3.0 alpha), including moving average filtering, and segmentation. Based on the estimated position and dimension of the target object, the end-effector pose (position and orientation) was calculated and fed to the custom IK module. The 3D point cloud-based approach is less dependent on diverse lighting conditions, as well as invariant to rotation as opposed to conventional approaches which usually require various poses of an object.

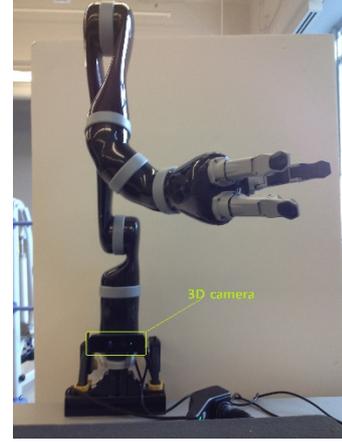


Figure 2. JACO with Senz3D camera

### B. Custom Inverse Kinematics Module

Before developing the custom IK module, we investigated two inherent problems with JACO. One problem is that the JACO API provides an IK function where the input is the center of the wrist-to-hand link, instead of the target end-effector pose. Because the default IK function does not consider the virtual link between the end-effector and the target object, there are inevitably collisions during object manipulation when default IK is used.

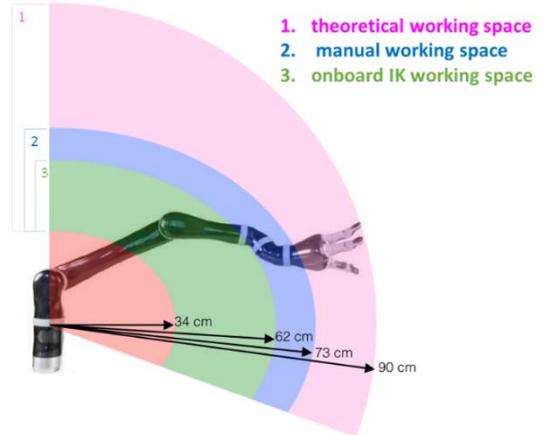


Figure 3. JACO working spaces

Another problem is the JACO workspace and positioning accuracy. According to the JACO technical documents, JACO can reach approximately 90cm in all directions using joystick control. However, when using the default IK function, we noticed that JACO has a decreased working space due to the embedded singularity avoidance algorithms. To examine the actual workspace where there is no limitation in performing translational, rotational and grasping motions, we programmed the JACO to automatically reach and perform all three basic motions (i.e., translational, rotational, and grasping) at 1cm resolution within the theoretical workspace of 90cm radius and 110 degrees of  $\phi$ .

(Figure 3). The JACO robot arm was found fully functional within the area 3 (i.e., a quarter-ellipsoid with about 62cm radius and 110 degrees of phi  $\phi$  excluding the dead zone). In the area 2 (a quarter-ellipsoid with around 73cm radius and 110 degrees of phi  $\phi$ ), one of the three basic motions does not work and JACO gets stuck due to the singularity avoidance algorithms, until manual control with the physical joystick overrides the current command. In the area 1, in addition to the same problem as in the area 2, the positioning accuracy of the JACO arm is severely compromised, and the end-effector has difficulty keeping still in place.

To address these issues, we developed a custom IK module that considered the missing tip-target link kinematic based on the target object pose through the 3D vision module and the robot parameters of the JACO robotic arm as shown in Table 1, 2, and 3. To compute minimum effort IK solution and plan trajectory to reach to a desired goal position, we adopted OpenRAVE's IKFast robot kinematics compiler, which analytically solves and generates optimized IK functions. We also adjusted dynamics caused by common factors like gravity and positioning tolerances by refining the IK solution using the Levenberg-Marquardt algorithm, also known as a damped least square method [18, 19]. The refined IK solution was sent to the JACO controller where virtual joystick signals emulating physical joystick commands were used to control JACO.

The arm sagging issue, that is the hand position of the JACO arm drops down 1-2cm whenever grasping commands are sent, was solved by using Cartesian command information (API function: GetCommandCartesianInfo()), instead of relying on the reported current arm position (API function: GetHandPosition()).

Table 1. D-H Parameters of JACO

DH Parameters				
	alpha(i-1)	a(i-1)	di	theta1
1	0	0	D1	q1
2	-pi/2	0	0	q2
3	0	D2	0	q3
4	-pi/2	0	d4b	q4
5	2*aa	0	d5b	q5
6	2*aa	0	d6b	q6

Table 2. Link length values

Link length values (meters)		
	Length	Explanation
D1	0.2102	Base to elbow
D2	0.4100	Arm
D3	0.2070	Front arm
D4	0.0750	First wrist
D5	0.0750	Second wrist
D6	0.1850	Wrist to the hand

Table 3. Alternate parameters

Alternate parameters	
aa	((11.0*PI)/72.0)
ca	(cos(aa))
sa	(sin(aa))
c2a	(cos(2*aa))
s2a	(sin(2*aa))
d4b	(D3 + (ca-c2a/s2a*sa)*D4)
d5b	(sa/s2a*D4 + (ca-c2a/s2a*sa)*D5)
d6b	(sa/s2a*D5 + D6)

### III. METHODS

#### A. Instruments

The instruments used for testing AROMA included a JACO robotic arm, a Senz3D camera, and a laptop computer running the custom software under Windows 7 Operating System. The JACO robotic arm was a research edition assembled in the year of 2012. All configuration parameters were maintained at default values throughout the experiment. The firmware version was 5.0.5.0033 and the API version was R5.0.2. The robot arm was fixed to a table using clamps and the Senz3D camera was attached to JACO 2cm below from the center of the robot base. The custom software, written in the C#/C++ programming language, controlled the 3D vision and the custom IK algorithms, while monitoring the robot behaviors.

#### B. Data Collection

In the experiment, picking-up/grasping tests were conducted with two different kinds of objects: balls with 3 different diameters (4.5cm, 6.5cm, and 8.5cm) and a bottle of water (Figure 4). The ball experiment started with commanding the robot to pick up a ball placed at a random position within JACO's theoretical workspace on the table. Upon successful grasping of the ball, the robot picked up the ball 50cm high and then dropped it off to a random position. The robot then went back to the default home position, and then automatically

repeated the same procedure 100 times. If the robot failed to grasp the ball in two consecutive times, the investigator put it in the random position manually. For the bottle experiment, the investigator manually put it at the random location (calculated by a random number generator) each time and the experiment was repeated 20 times. Throughout the tests, the estimated object locations and dimensions were recorded, and performance measures including time to grasp the target object (from the start of each trial to when the target object is picked up 50cm high) and success rate were calculated.



Figure 4. Test Objects

#### IV. RESULTS

The results from the experiments are presented in Table 4.

Table 4. Test Results

Ball Size	Ball Experiment		Bottle Experiment	
	Average Grasping Time (sec)	Success Rate	Average Grasping Time (sec)	Success Rate
S	5.51 ( $\pm 1.38$ )	93/100	5.96 ( $\pm 1.95$ )	17/20
M	4.17 ( $\pm 0.97$ )	100/100		
L	4.46 ( $\pm 1.48$ )	100/100		

From Table 1, it took longer to grasp the bottle than the balls, possibly due to different grasping strategies (Figure 5). Based on the estimated dimension of the target object, either grasping from the side or the top will be automatically selected.

As for the success rate, grasping the small ball was least reliable with 93% success rate. In the failed trials, the JACO hand was able to pick up the ball but then dropped it off before reaching to the target height. We speculated that the glossy surface of the small ball might compromise the object pose estimation and thus lead to unreliable grasping points. Table 5 shows the deviations between actual object widths and estimated ones for each ball. The small ball not only had the largest variance among

the three balls, but also had tendency to underestimate the object size. The success rate of the bottle grasping test was 85%. The failed trials were mostly due to the collisions between the JACO hand and the bottle. For both the small ball and bottle experiments, the object locations were well distributed and the locations (marked in red) where unsuccessful trials occurred were highly scattered and no systemic pattern was found (Figure 7 and 8). In addition to the object pose estimation error, JACO's positioning tolerance of  $\pm 8\text{mm}$  might also affect the performance.



Figure 5. Different grasping strategies

Table 5. Object Width Estimation

Width (mm)	Small	Medium	Large
Actual	45	65	85
Estimate	39.1( $\pm 4.57$ )	64.3( $\pm 2.57$ )	84.86( $\pm 2.28$ )

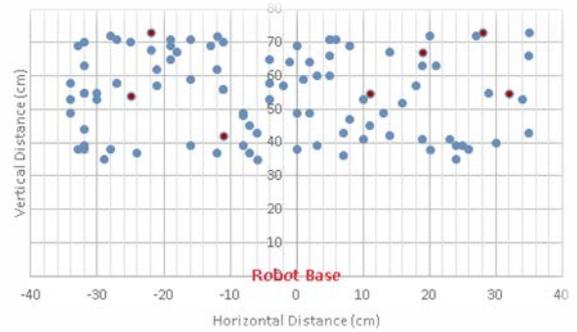


Figure 6. Small Ball Placements (cm)

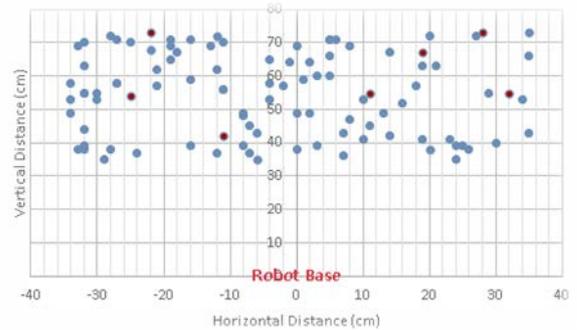


Figure 7. Bottle Placements (cm)

## V. DISCUSSION

Our results indicated that AROMA has potential to enable users who are currently unable to use an assistive robotic manipulator to use it by providing an autonomous or semi-autonomous robotic manipulation assistance. The AROMA has some advantages over the conventional vision-based approach. First, unlike other research studies [7-9, 12, 13], it relies on point clouds generated from a low-cost 3D depth sensing camera, thus the computational cost is less than the conventional 3D object pose estimation algorithms which require images of various poses such as the front side, backside, and all possible 3D rotations of the object. Chung and colleagues evaluated a vision-based autonomous function of assistive robotic manipulator, mounting a high resolution webcam on the robot shoulder [12]. In the study, they measured the task completion time and the success rate for a drinking task which consists of various subtasks including picking up the drink from a start location, conveying the drink to the proximity of the user's mouth. The average task completion time for picking up a soda can on the table was 12.55 ( $\pm 2.72$ ) seconds including the average object detection time of 0.45 ( $\pm 0.12$ ) seconds. The success rate of the pick-up task was 70.1% (44/62).

Second, the AROMA uses infrared images, and thus is less dependent on ambient lighting conditions than conventional image processing which requires images of an object under different lighting conditions or sources in order to improve the algorithm invariance to diverse lighting conditions. Tsui et al. developed a vision-based autonomous system for a wheelchair-mounted robotic manipulator using two stereo cameras, one mounted over the shoulder on a fixed post and one mounted on the gripper. Once the user only needed to indicate the object of interest by pointing to the object on a touch screen, the autonomous control automatically took over the rest of the task by reaching towards the object, grasping it, and bringing it back to the user [10]. They evaluated this system with 12 individuals with various physical and cognitive disabilities, where participants were asked to retrieve an object from a bookshelf. The success rate of the autonomous function was 65% (129/198). Of the 69 unsuccessful trials, 56 (81%) were due to algorithm failures. Jiang et al. also developed a vision-based autonomous robot control system combining a JACO robot arm with two Microsoft Kinect sensors: one for recognizing user voice, gesture and body part; the other for object recognition [11]. User's voice and hand gestures were used as the robot control

commands. The object recognition algorithm relied on a two-step process, which extracted the feature vector for an object using Histogram of Oriented Gradients algorithm, then trained the model and classified the objects applying nonlinear support vector machine algorithm. The system was evaluated by one participant with four different manipulation tasks (5 trials per each), including, drinking, phone calling, taking a self-portrait, and taking photos of the surroundings. The performance time ranged from 14-130 seconds and accuracy ranged from 52-98%.

Third, AROMA addressed the inherent limitations of the JACO onboard IK algorithm, including the missing tip-target link, reduced working space, and arm sagging issues. In addition, the AROMA was developed under a Windows operating system, making it not only easier to integrate new and existing alternative input devices without developing additional driver software, but also increasing the likelihood of adoption by users and clinical professionals.

However, AROMA has also several limitations. First, when dealing with the missing tip-target link, we aimed to find a goal configuration of the end-effector that matches the target object pose under the assumption that there is no obstacle between the manipulator and the target object. This may compromise the manipulation performance and safety in challenging environments such as cluttered space. To address this issue, additional sensors such as an eye-in-hand camera or force/tactile sensors could be adopted. Second, the experiments were conducted with simple shaped objects with smooth surfaces. To accommodate a variety of everyday objects with different characteristics, the damping factor for the DLS method may need to be adjusted to achieve a balance between performance stability and speed. Lastly, it is also important to apply AROMA to real-world manipulation tasks and test it with individuals having upper extremity impairments.

In addition to supporting autonomous operation of JACO, a practical application of AROMA is to support semi-autonomous control, where direct user control is combined with robot autonomy, strategically reducing the complexity exposed to the user while keeping the user in the control loop [20]. Users usually find fine manipulation of a robot manipulator more challenging and spend more time on adjusting the end-effector position and orientation before grasping. AROMA could potentially address this issue by allowing users to use conventional input methods (e.g., joystick) to move the arm close to the

target object, and then user voice control to command the robot for fine manipulation, e.g., grasping or pushing. Kim et al. found that while user effort required for operating the robot with autonomous control was significantly less than with the manual control, user satisfaction with the autonomous control was lower than with the manual control [13]. With the semi-autonomous control, users only need to control the gross motion and leave the fine manipulation to AROMA, which could potentially lead to improved performance and satisfaction.

We are planning on two follow-up studies to apply the AROMA. One study is to apply the semi-autonomous approach to an overhead track mounted assistive robotic system called KitchenBot [21], which operates along an overhead track built into the kitchen to assist individuals with physical disabilities for tasks in a typical kitchen environment. Another study is to combine AROMA with automatic speech recognition to provide complete hands-free semi-autonomous operation.

#### ACKNOWLEDGMENT

This work is supported by Craig H. Neilsen Foundation (#338434) and with resources and use of facilities at the Human Engineering Research Laboratories (HERL), VA Pittsburgh Healthcare System. This material does not represent the views of the Department of Veterans Affairs or the United States Government.

#### REFERENCES

- [1] S. Allin, E. Eckel, H. Markham, and B. R. Brewer, "Recent trends in the development and evaluation of assistive robotic manipulation devices," *Physical Medicine and Rehabilitation Clinics of North America*, vol. 21, pp. 59-77, 2010.
- [2] G. Romer, H. J. Stuyt, and A. Peters, "Cost-savings and economic benefits due to the assistive robotic manipulator (ARM)," in *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on*, 2005, pp. 201-204.
- [3] V. Maheu, J. Frappier, P. Archambault, and F. Routhier, "Evaluation of the JACO robotic arm: Clinico-economic study for powered wheelchair users with upper-extremity disabilities," in *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, 2011, pp. 1-5.
- [4] G. W. Römer, H. Stuyt, G. Peters, and K. van Woerden, "14 Processes for Obtaining a "Manus"(ARM) Robot within the Netherlands," in *Advances in Rehabilitation Robotics*, ed: Springer, 2004, pp. 221-230.
- [5] G. Romer, H. J. Stuyt, and A. Peters, "Cost-savings and economic benefits due to the assistive robotic manipulator (ARM)," in *Proceedings of the 2005 IEEE, 9th International Conference on Rehabilitation Robotics*, 2005, pp. 201-204.
- [6] C.-H. King, T. L. Chen, Z. Fan, J. D. Glass, and C. C. Kemp, "Dusty: an assistive mobile manipulator that retrieves dropped objects for people with motor impairments," *Disability and Rehabilitation: Assistive Technology*, vol. 7, pp. 168-179, 2012.
- [7] B. Driessen, T. T. Kate, F. Liefhebber, A. Versluis, and J. van Woerden, "Collaborative control of the manus manipulator," *Universal Access in the Information Society*, vol. 4, pp. 165-173, 2005.
- [8] H. A. Tijsma, F. Liefhebber, and J. Herder, "Evaluation of new user interface features for the manus robot arm," in *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on*, 2005, pp. 258-263.
- [9] I. Laffont, N. Biard, G. Chalubert, L. Delahoche, B. Marhic, F. C. Boyer, *et al.*, "Evaluation of a graphic interface to control a robotic grasping arm: a multicenter study," *Archives of physical medicine and rehabilitation*, vol. 90, pp. 1740-1748, 2009.
- [10] K. M. Tsui, D.-J. Kim, A. Behal, D. Kontak, and H. A. Yanco, "'I want that': Human-in-the-loop control of a wheelchair-mounted robotic arm," *Applied Bionics and Biomechanics*, vol. 8, pp. 127-147, 2011.
- [11] H. Jiang, T. Zhang, and J. P. Wachs, "Autonomous Performance of Multistep Activities with a Wheelchair Mounted Robotic Manipulator Using Body Dependent Positioning," 2014.
- [12] C.-S. Chung, H. Wang, and R. Cooper, "Autonomous function of wheelchair-mounted robotic manipulators to perform daily activities," in *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, 2013, pp. 1-6.
- [13] D.-J. Kim, R. Lovelett, and A. Behal, "Eye-in-hand stereo visual servoing of an assistive

- robot arm in unstructured environments," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, 2009, pp. 2326-2331.
- [14] S. S. Srinivasa, D. Ferguson, C. J. Helfrich, D. Berenson, A. Collet, R. Diankov, *et al.*, "HERB: a home exploring robotic butler," *Autonomous Robots*, vol. 28, pp. 5-20, 2010.
- [15] H. Tanaka, Y. Sumi, and Y. Matsumoto, "Assistive robotic arm autonomously bringing a cup to the mouth by face recognition," in *Advanced Robotics and its Social Impacts (ARSO), 2010 IEEE Workshop on*, 2010, pp. 34-39.
- [16] P. I. Corke, *Visual Control of Robots: high-performance visual servoing: Research Studies Press Taunton, UK*, 1996.
- [17] D. Modarress, P. Svitek, K. Modarress, and D. Wilson, "Micro-Optical Sensors for Boundary Layer Flow Studies," in *ASME 2006 2nd Joint US-European Fluids Engineering Summer Meeting Collocated With the 14th International Conference on Nuclear Engineering*, 2006, pp. 1037-1044.
- [18] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 16, pp. 93-101, 1986.
- [19] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *Journal of dynamic systems, measurement, and control*, vol. 108, pp. 163-171, 1986.
- [20] C. C. Kemp, A. Edsinger, and E. Torres-Jara, "Challenges for robot manipulation in human environments," *IEEE Robotics and Automation Magazine*, vol. 14, p. 20, 2007.
- [21] D. Ding, H. W. Ka, R. Cooper, J. Telson, and K. Krishnaswamy, "Focus Group Evaluation of an Overhead Kitchen Robot Appliance," presented at the RESNA, Indianapolis, IL, 2014.