Evaluation of an Inverse-Kinematics Depth-Sensing Controller for Operation of a Simulated Robotic Arm

Akhilesh Kumar Mishra[1], Lourdes Peña-Castillo[1,2], and Oscar Meruvia-Pastor[1(✉)]

[1] Department of Computer Science, Memorial University of Newfoundland, St John's, Canada
{Akm565,Lourdes,Oscar}@mun.ca
[2] Department of Biology, Memorial University of Newfoundland, St John's, Canada

Abstract. Interaction using depth-sensing cameras has many applications in computer vision and spatial manipulation tasks. We present a user study that compares a short-range depth-sensing camera-based controller with an inverse-kinematics keyboard controller and a forward-kinematics joystick controller for two placement tasks. The study investigated ease of use, user performance and user preferences. Task completion times were recorded and insights on the measured and perceived advantages and disadvantages of these three alternative controllers from the perspective of user efficiency and satisfaction were obtained. The results indicate that users performed equally well using the depth-sensing camera and the keyboard controllers. User performance was significantly better with these two approaches than with the joystick controller, the reference method used in comparable commercial simulators. Most participants found that the depth-sensing camera controller was easy to use and intuitive, but some expressed discomfort stemming from the pose required for interaction with the controller.

Keywords: Gesture-based controllers for robot arm manipulation · Depth-sensing cameras and short-range RGBD sensors · Inverse and forward kinematics · User studies

1 Introduction

Human-Computer Interaction using depth-sensing cameras has captured the imagination of enthusiasts and researchers alike. Depth-sensing cameras are especially suited for spatial manipulation tasks, and are nowadays used in many areas, including mobile computing, gaming and robotics [1–4]. After the introduction of the first Kinect, a new group of depth-sensing cameras with a much closer range of interaction appeared in the market, allowing for the technology to be used in a wider variety of applications in desktops, laptops, wearable and mobile devices [5–9].

In this article, we investigate a depth-sensing camera controller based on inverse-kinematics for placement tasks using a robot arm simulator, and describe a user

study to evaluate participants' performance using short-range depth-sensing camera controllers against comparable off-the-shelf controllers.

2   Related Work

In recent years there has been a significant increase in the usage of robotic arms that require human manipulation in industrial, medical and offshore applications.

One of the earliest manipulation problems studied in the field of robotics was the insertion of a peg into a hole using a robotic arm while preventing the wedging or jamming of the peg [10]. Since then, there have been significant advances in manipulation techniques and robot control. There are many techniques for manipulating a robot arm, such as the Titan IV robot arm, involving position-controlled manipulators [11–13], joystick based controllers [11, 14], speech [15, 16], gesture based controllers [2, 3, 17], computer simulations [18, 19], and even smartphones [20]. To help training operators to control a robotic arm, there are virtual simulators like GRI Simulations Inc.'s VROV manipulator trainer [19], which help a user train on a particular robotic arm type using either a master controller or a joystick. Compared to joysticks, master controllers are more sophisticated and expensive devices. On the other hand, joysticks are more affordable, but not as convenient as master controllers, since they are generic use products designed for gaming which do not exactly map to the functionalities of master controllers.

In this research, we evaluated alternatives to control robots arms using off-the-shelf input devices, such as cameras and keyboards. Previous methods which use either stereo cameras or depth-sensing cameras to control a robot are described in [2–4]. To manipulate the simulated robotic arm used in this study, we implemented the method presented by Mishra et al. [21]. In that approach, the user specifies the target position of the robot actuator by moving his/her hands in front of a depth-sensing camera. The depth camera returns the coordinates of the users' hand and the coordinates are passed as target position to an inverse kinematics module, where the joint angles for the arm simulator are calculated using CCD [22]. After the joint angles have been calculated, the robot arm simulator module applies the rotations and the end-effector reaches the target.

3   Methods

We compared user performance using the inverse-kinematics control method proposed by Mishra et al. [21] to control a robot-arm of type Titan IV (Fig. 1a) against user performance using two other common control devices: keyboards and joysticks. Keyboards were selected for comparison because most users are familiar with the use of keyboards, while the joystick controller is the reference interface used in some commercial simulators. Master controllers, such as the one shown in Fig. 1b, were not included in this study because they are costly (in the range of tens of thousands of dollars) and highly specialized devices not usually available to most users.
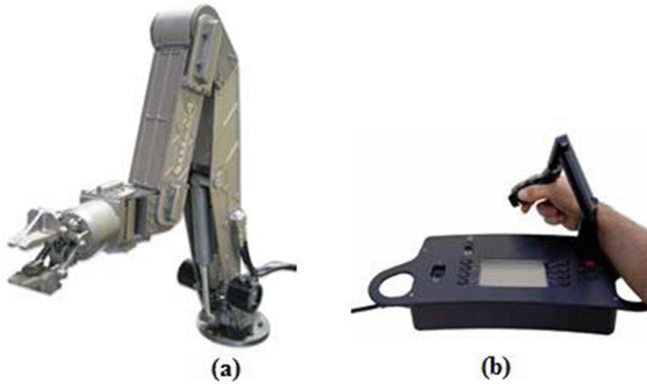
Fig. 1. (a) Titan 4 robotic arm (b) Master controller for Titan 4 robotic arm. Image credit: Copyright 2005FMC Technologies, Inc.

The goal of the study was to find answers to three main questions: (1) How will users' performance with a depth-sensing camera control system compare to their performance using the other alternatives? (2) How long do users needed to get used to each method? (3) Which controller will users prefer the most?. To answer the above mentioned questions and to evaluate in more detail the user's impressions on the different type of controllers, a robot arm simulator shown in Fig. 2 was developed to carry out the study.
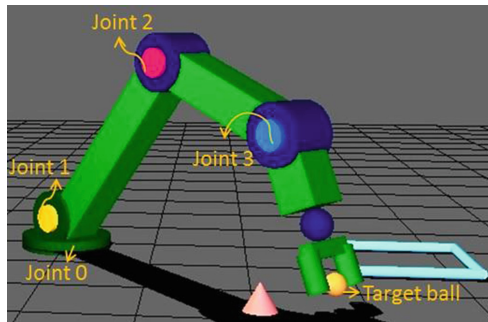

Fig. 2. Robot arm simulated with OpenGL.

3.1   User Study

Our user study had a balanced design with 18 participants recruited, 2 tasks, 3 controllers and 6 orderings of controller usage. An ordering of controller usage, for example, was to use first the keyboard controller, second the depth camera controller, and third the joystick controller. Each user performed each task 10 times with each controller in the assigned order of usage. Their task completion times were recorded and their feedback with regard to the different controllers was collected with an exit questionnaire.

4   Results

4.1   Analysis of Users' Performance

Table 1 shows the mean execution times per controller for each task. Using the Shapiro-Wilk test and Bartlett's test, we determined that our data significantly deviated from normality and had heterogeneous variances. Thus, we performed non-parametric tests, such as the Kruskal-Wallis and the Friedman test, in addition to two-way ANOVA to test whether the average completion time per task differed between controllers when considering the ordering of usage. The conclusions were the same irrespective of the statistical test used. Statistical analysis was done using R (version 3.1.1).

Table 1. Mean execution time per controller (secs.)

Controller	Keyboard	Camera	Joystick
Task 1	17.3 ± 14.5	18.1 ± 11.9	28.7 ± 12.9
Task 2	29.6 ± 17.5	30.8 ± 13.4	38.0 ± 17.6

Figure 3 shows the completion times per controlling device per task. The average completion time for both tasks varied significantly depending on the controller used (p-value $< 2e-16$ and p-value $= 6.78e07$ for Task 1 and Task 2 respectively).
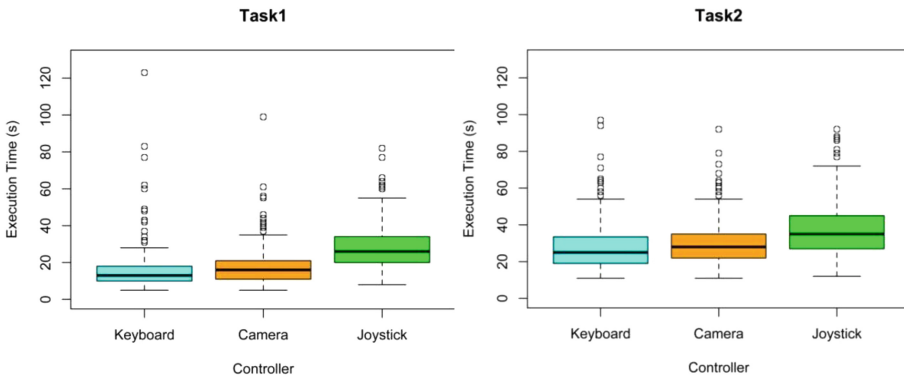

Fig. 3. Execution times per controller for Task 1 and Task 2. The black lines inside the boxes indicate the median completion time, the bottoms of the boxes are at the 25 percentile and the tops are at the 75 percentile.

The results indicate that the depth-sensing camera controller and the keyboard controller allowed for similar levels of user performance. User performance with these two approaches was significantly better than their performance using the joystick controller, which is the reference method used in comparable commercial simulators. Our results also show that there is a significant learning effect in the case of the depth-camera controller, which was also perceived by the participants (see Fig. 4).
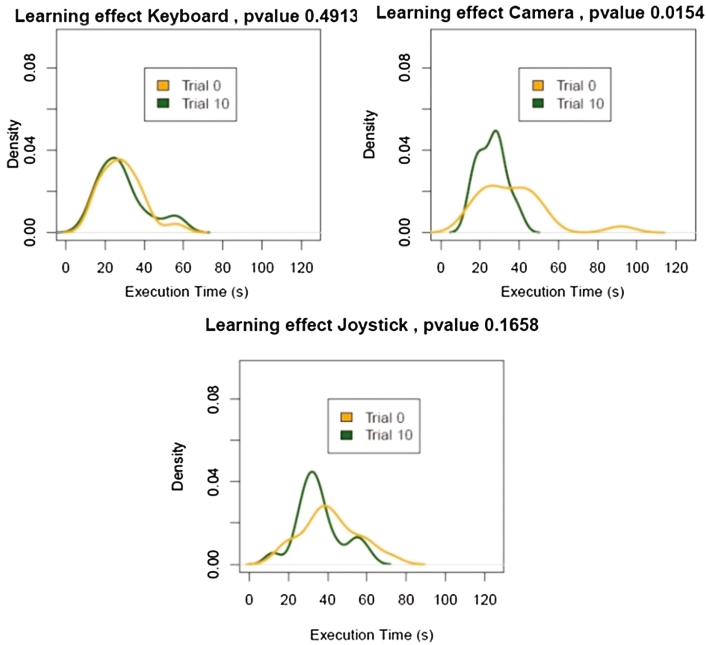
Fig. 4. Learning effect for each input device for Task 1 and Task 2 (Color figure online)

4.2   User Perception and Preference

This section discusses the data gathered from the exit questionnaire.

Participants' perception on ease of use, ease of learning, usefulness, ergonomics, performance and overall preference for each controller was collected. Participants found that the depth-sensing camera controller was both easy to use and intuitive, but expressed some discomfort during interaction. This discomfort is a known usability problem of vision-based gesture interactions [23].

Perception of Ease of Learning. The majority of participants found the three controllers easy to learn, with 15 participants (83.33 %) strongly agreeing that the depth-camera method was easy to learn, 14 participants (77.78 %) strongly agreeing that the keyboard method was easy to learn, and 12 participants (66.66 %) strongly agreeing that the joystick method was easy to learn. The complete results about the perception of ease of learning are shown in Fig. 5.

Perception of Performance. No participant found the performance of the controllers inappropriate (Fig. 6). More than 88 % of the participants agree that the performance of the depth-camera and keyboard controllers was appropriate.

Perception of Ease of Use. Figure 7 shows the user responses about whether each input device was easy to use. Users rated the depth-camera and keyboard as easy to use. The joystick was rated slightly lower as compared to other input devices. In general, users found all the devices easy to use.
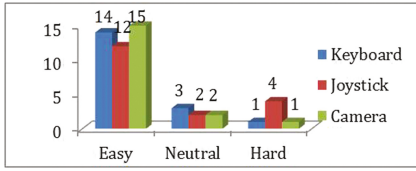
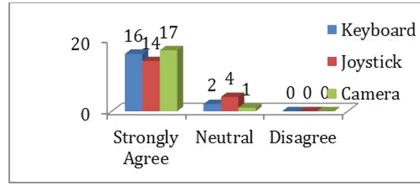Fig. 5. Perception of ease of learning (Color figure online)

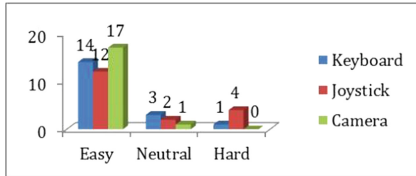Fig. 6. Perception of performance (Color figure online)

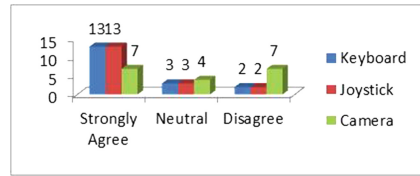Fig. 7. Perception of ease of use (Color figure online)

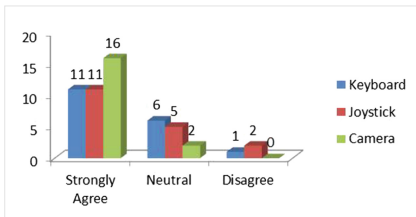Fig. 8. Ergonomics rating of each controller (Color figure online)

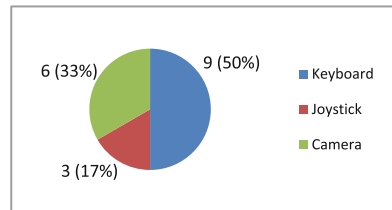Fig. 9. Usefulness rating of each controller (Color figure online)

Fig. 10. User responses for least amount of training time (Color figure online)

Perception of Ergonomics. The depth-camera interface was rated lower in ergonomics ratings when compared to both the keyboard and the joystick interface. Figure 8 shows the user responses about ergonomics of each input device. Only 7 out of 18 participants reported that the depth-camera was an ergonomic interface and some of the unstructured user feedback (discussed below) provided some further insight into this result.

Perception of Usefulness. The depth-camera interface was rated highest in usefulness when compared to both the keyboard and the joystick interface (Fig. 9), with 16 out of 18 participants strongly agreeing that the depth-camera was a useful interface with the rest being neutral about it, whereas the keyboard and joystick controllers were found useful by 11 participants.

Perception of Training Time. Figure 10 illustrates that 9 participants (50 %) believed that the keyboard took the least amount of training time. The depth-camera was rated second in training time, with 6 participants (33 %) saying that the depth-camera
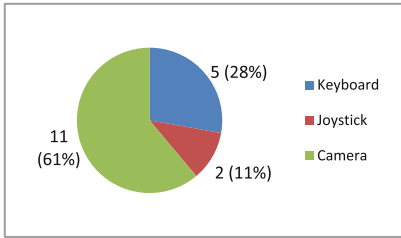
Fig. 11. Users' perception of performance after training (Color figure online)
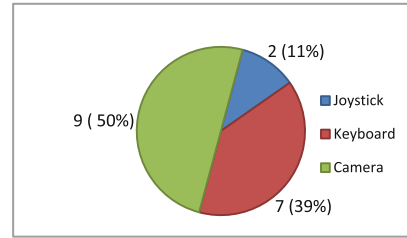

Fig. 12. Overall input controller preference (Color figure online)

requires the least amount of training. The joystick was rated lowest as only 3 participants (17 %) said that joystick requires the least amount of training. We believe this impression was caused because the joystick was controlled using forward kinematics, making it harder to use than the other alternatives.

Perception of Performance after training. Figure 11 shows that 11 (61 %) participants reported that the depth-camera performs best after training, 5 (28 %) participants reported that keyboard performs best and only 2 (11 %) participants chose the joystick as best performer after training and practice. It is interesting to note that users favored the depth-camera controller after training, even though they performed equally well with the keyboard.

Overall Preference. Figure 12 shows user's overall controller preference. 9 participants (50 %) responded that they preferred the most the depth-camera controller. The second most preferred controller by 7 participants (39 %) was the keyboard. The least preferred option is the joystick controller which is the controller preferred by 2 participants (11 %).

Written User Feedback. The questionnaire also asked participants to write comments about the experiment and the interface. With respect to the depth camera position, some users reported that the camera position played an important role in overall performance of the depth-camera controller. The depth-camera was mounted on top of the desktop monitor like a webcam and, for some users, that was not a convenient position. One participant suggested that, instead of mounting the depth-camera on top of the monitor, the depth-camera should be kept beside the keyboard on the same plane as the keyboard, facing upwards so that the user does not have to lift their hand too high to interact with the camera. Also, another user suggested that there should be a support for the elbow if the camera was to be mounted on top of the desktop monitor. Causing discomfort is a known usability problem of vision-based gesture interaction systems and there are steps that can be taken to reduce fatigue, as proposed by [23]. Another alternative to reduce fatigue is to implement a hand gesture control system providing tactile feedback to the user's hand, such as the one proposed by Kim et al. [24].

5   Conclusion

We comparatively assessed three different devices to control a robotic arm, namely, depth-sensing camera, keyboard and joystick. To do this, we developed a robot arm simulator which could be controlled using a standard keyboard, a gaming joystick and a depth-sensing camera. In addition, we conducted a user study where 18 participants were recruited to participate. Our results indicate that the task completion times for the depth-camera and the keyboard controllers were significantly lower than the joystick controller, without any significant statistical difference observed between the depth-camera and the keyboard. Our results also show that there is a significant learning effect in the case of the depth-camera controller which was also perceived by the participants. While participants had a positive perception of all three controllers, 50 % of the users reported that they would prefer the depth-camera interface over the joystick and the keyboard interface. The reason why the joystick was least preferred could be related to the fact that the joystick was using forward kinematics and its execution times are significantly longer, compared to the depth-camera and the keyboard.

References

1. Lun, R., Zhao, W.: A Survey of Applications and Human Motion Recognition with Microsoft Kinect. Int. J. Pattern Recogn. Artif. Intell. (2015)
2. Corradini, A., Gross, H.: Camera-based gesture recognition for robot control. In: 2000 Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000 (2000). doi:10.1109/IJCNN.2000.860762
3. A. Anonymous: Real-time 3D pointing gesture with kinect for object-based navigation by the visually impaired. Presented at Biosignals and Biorobotics Conference (BRC), 2013 ISSNIP (2013). doi:10.1109/BRC.2013.6487535
4. Suay, H.B., Chernova, S.: Humanoid robot control using depth camera. In: 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (2011)
5. 31 October 2014. http://www.roadtovr.com/nimble-sense-kickstarter-virtual-reality-oculus-rift-time-of-flight-input/
6. Leap Motion Controller, 10 June 2015. https://www.leapmotion.com/
7. Microsoft Kinect for Windows V2, 25 June 2015. https://www.microsoft.com/en-us/kinectforwindows/
8. Intel Real Sense SDK, 16 June 2015. https://software.intel.com/en-us/intel-realsense-sdk
9. Project Tango, ATAP, 25 June 2015. https://www.google.com/atap/project-tango/
10. Whitney, D.E.: Quasi-static assembly of compliantly supported rigid parts. J. Dyn. Syst. Measur. Control **104**(1), 65–77 (1982)
11. Sian, N.E., Yokoi, K., Kajita, S., Kanehiro, F., Tanie, K.: Whole body teleoperation of a humanoid robot - development of a simple master device using joysticks. In: 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (2002). doi:10.1109/IRDS.2002.1041657
12. Manikandan, R., Arulmozhiyal, R.: Position control of DC servo drive using fuzzy logic controller. In: 2014 International Conference on Advances in Electrical Engineering (ICAEE) (2014). doi:10.1109/ICAEE.2014.6838474

13. Jan, V., Marek, S., Pavol, M., Vladimir, V., Stephen, D.J., Roy, P.: Near-time-optimal position control of an actuator with PMSM. In: 2005 European Conference on Power Electronics and Applications (2005). doi:10.1109/EPE.2005.219516

14. Lu, Y., Huang, Q., Li, M., Jiang, X., Keerio, M.: A friendly and human-based teleoperation system for humanoid robot using joystick. In: 2008 7th World Congress on Intelligent Control and Automation, WCICA 2008 (2008). doi:10.1109/WCICA.2008.4593278

15. Ng-Thow-Hing, V., Luo, P., Okita, S.: Synchronized gesture and speech production for humanoid robots. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2010)

16. House, B., Malkin, J., Bilmes, J.: The VoiceBot: a voice controlled robot arm. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (2009). http://doi.acm.org/10.1145/1518701.1518731. doi:10.1145/1518701.1518731

17. Oniga, S., Tisan, A., Mic, D., Buchman, A., Vida-Ratiu, A.: Hand postures recognition system using artificial neural networks implemented in FPGA. In: 30th International Spring Seminar on Electronics Technology (2007). doi:10.1109/ISSE.2007.4432909

18. Chen, C.X., Trivedi, M.M., Bidlack, C.R.: Simulation and animation of sensor-driven robots. IEEE Trans. Robot. Autom. **10**(5), 684–704 (1994). doi:10.1109/70.326572

19. (18/06/2015). VROV Manipulator Trainer [VROV Manipulator Trainer]. http://www.grisim.com/products/vrov-virtual-remotely-operated-vehicle/

20. Parga, C., Li, X., Yu, W.: Tele-manipulation of robot arm with smartphone. In: 2013 6th International Symposium on Resilient Control Systems (ISRCS) (2013). doi:10.1109/ISRCS.2013.6623751

21. Mishra, A.K., Meruvia-Pastor, O.: Robot arm manipulation using depth-sensing cameras and inverse kinematics. Oceans - St. John's (2014). doi:10.1109/OCEANS.2014.7003029

22. Kenwright, B.: Inverse kinematics - cyclic coordinate descent (CCD). J. Graphics, GPU & Game Tools, 177–217 (2012)

23. Hincapié-Ramos, J.D., Guo, X., Moghadasian, P., Irani, P.: Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. Presented at Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. 2014, Available: http://doi.acm.org/10.1145/2556288.2557130. doi:10.1145/2556288.2557130

24. Kim, K., Kim, J., Choi, J., Kim, J., Lee, S.: Depth camera-based 3D hand gesture controls with immersive tactile feedback for natural mid-air gesture interactions. Sensors (Basel) **15**(1), 1022–1046 (2015). doi:10.3390/s150101022