EchoBot: Facilitating Data Collection for Robot Learning with the Amazon Echo

Rishi Kapadia, Sam Staszak, Lisa Jian, Ken Goldberg
The AUTOLAB at UC Berkeley

***Abstract*— The Amazon Echo and Google Home exemplify a new class of home automation platforms that provide intuitive, low-cost, cloud-based speech interfaces. We present EchoBot, a system that interfaces the Amazon Echo to the ABB YuMi industrial robot to facilitate human-robot data collection for Learning from Demonstration (LfD). EchoBot uses the computation power of the Amazon cloud to robustly convert speech to text and provides continuous speech explanations to the user of the robot during operation. We study performance with two tasks, grasping and "Tower of Hanoi" ring stacking, with four input and output interface combinations. Our experiments vary speech and keyboard as input interfaces, and speech and monitor as output interfaces. We evaluate the effectiveness of EchoBot when collecting infrequent data in the first task, and evaluate EchoBot's effectiveness with frequent data input in the second task. Results suggest that speech has potential to provide significant improvements in demonstration times and reliability over keyboards and monitors, and we observed a 57% decrease in average time to complete a task that required two hands and frequent human input over 22 trials.**

I. Introduction

With the emergence of voice activation systems, a new class of home automation platforms has appeared in the commercial market. These systems utilize speech recognition and natural language processing to facilitate interactions with a variety of devices, including speakers, smartphones, and television sets. Speech interfaces also have potential to enhance the efficiency of interactions with robots, such as to train a robot to perform a task according to a desired policy. One approach to training robots is Learning from Demonstration (LfD), where a human provides several demonstrations of a task to the robot, and the robot learns to perform that task. These demonstrations may consist of segments of arm trajectories or keyframes of robot poses at periodic time intervals, and are specified to the robot as input. The robot uses this collected data to learn a policy to perform the task.

We explore how a voice activation system may improve data collection for LfD. Using the Amazon Echo [4], we implemented EchoBot[1,2], a 2-way speech interface for communication between humans and robots during data collection for robot learning.

EECS & IEOR, University of California, Berkeley, CA, USA; {rishikapadia, samstaszak, lisajian, goldberg}@berkeley.edu

[1]Code is available at https://github.com/rishikapadia/echoyumi.

[2]Video is available at https://www.youtube.com/watch?v=XgaGeCsERU8.

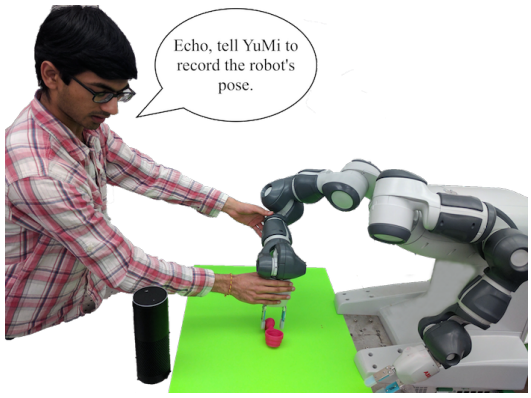

Fig. 1: EchoBot integrates the Amazon Echo home automation bi-directional speech interface with the ABB YuMi industrial robot. We present results comparing EchoBot to a keyboard-monitor input-output interface.

A. System Criteria

In this paper, we focus on a speech interface as a medium for input and output. We assume that there is only one person speaking during the command transmission and that the user knows what keywords issue each command. To be usable and convenient to the user, response times to user commands must have a latency similar to the delay between two conversing humans.

B. System Overview

We integrated the Amazon Echo and the YuMi industrial robot from ABB [1] into EchoBot. The Echo is a wireless speaker and voice command device. It is a low-cost, commercially-available product with a text-to-speech interface for natural language processing. We use the Echo for its speech recognition system, which is robust to variances in voice location, pitch, and intonation. The YuMi is a dual-arm, human-safe industrial robot with flexible joints and grippers, and offers state-of-the-art robot control. EchoBot as a system allows users to utter commands to the Echo that are relayed as actions to the YuMi robot, and communicates vocal feedback from the robot back to the user while performing those actions.

We evaluate EchoBot in 2 experiments as an input and output interface to perform data collection using robots and find that it increases collection efficiency when the user needs to input data frequently while both hands are occupied with the task.

This paper contributes:

1) The first implemented system architecture interfacing the Amazon Echo home automation speech interface to

the ABB YuMi industrial robot.
2) Experiments with two robot tasks, grasping and "Tower of Hanoi" ring stacking, comparing four interface combinations varying input and output with keyboard, monitor, and speech.

II. Related Work

A. Data Collection for Learning from Demonstration

LfD is a promising approach to teach policies to robots through demonstrations of a desired behavior. It can take robot trajectories and data labels as input, and a policy function as output. There currently exist several methods for providing demonstrations to robots, including teleoperation and kinesthetic teaching [5]. In teleoperation, the demonstrator controls the end-effector position or joint angles of the robot using a device such as a joystick or game controller [25]. In kinesthetic teaching, the human physically guides the robot's arms and grippers to complete the task. The robot uses several such demonstrations of the task as a sampling of the policy intended by the human, and attempts to find the underlying policy to perform the task. Both data collection methods often require the demonstrator to use both hands, which can complicate denoting the start and end of a demonstration using a game controller or button press.

Several studies have utilized voice commands to facilitate data collection of kinesthetic demonstrations in LfD systems. In [43], subjects were tasked with using voice commands to start and end demonstrations, and afterward the robot reproduced the learned skill. The speech interface of Akgun et. al [2] had similar functionality. The kinesthetic trajectories were provided in keyframes, where voice commands were used to indicate where the demonstration was segmented by the subject. In addition to keyframe and segmentation functionality, EchoBot also provides the user with prompts and continuous audio output detailing the status of the robot.

B. Home Automation Systems

The Echo has been used as an interface for products by Uber, StubHub, Fitbit, Domino's Pizza, and many others [3]. Samsung revealed in late 2016 that all of its WiFi-enabled robotic vacuums can now be controlled using the Echo. Other voice interfaces include Apple's Siri in 2011 and Homekit in 2014, a collection of smart devices for users to control around the house. In November 2016, Google Home was introduced, which offers the capability of Google, Inc.'s search engine. Other smart assistants include ivee Voice, Cubic, Mycroft, Sonos Play, Hal, Comcast Xfinity TV remote, and many others.

Prior work has studied voice activation to control robots and other machines. An early instance of voice recognition used in surgical robotic assistants controlled the end effector location of a robotic arm during laparoscopic surgery [34], where the surgeon could command the arm to move in the 3 axial directions or to predefined locations at a constant speed. Furthermore, the constraints or failure modes of the robot were conveyed audibly to the surgeon, such as when a joint had exceeded its limits, to prevent damage to the robot and patient. Dean et al. [10] used a speech interface to control the da Vinci, a tele-operation robotic surgical system, to perform simple tasks like measuring the distance between two locations and manipulating visual markers on the display. Henkel et al. [19] describes an open-source voice interaction toolkit to serve as a medium between dependent victims, such as trapped earthquake survivors, and the outside world. Gesture and voice interfaces were developed to help disabled people operate a remote controller for home automation [21], to facilitate rehabilitation for people with disabilities [26], and to help children with autism [33]. Other examples of using voice control for robots include [16, 18, 29, 42]. To our knowledge, EchoBot is the first of these home automation systems that provides voice interaction to an industrial robot for data collection.

C. Robots and Speech

Our motivation for facilitating data collection using a voice interface comes from the work of previous studies. Ray et al. [32] found that humans prefer to interact with robots using speech. Cha et al. [8] have shown that human perception of robot capability in physical tasks can be affected by speech. Takayama et al. [37] found that perception of robots is influenced not only by robots showing forethought, but also by the success outcome of the task and showing goal-oriented reactions to task outcomes. The acceptance of robots is important in making robots a part of workplaces and homes [6, 9], and the perceived capability of robots largely influences robot acceptance [9]. Srinivasa et al. [36] used a speech synthesis module on their robotic butler to interact with humans while completing tasks such as collecting mugs. When humans engaged with robots using speech, their confidence that the robot was a reliable source of information was shown to increase [28]. It has also been shown that there are noticeable drops in trust as reliability of the robot decreases [11, 13, 14], and only once reliability recovers does trust start to increase monotonically [12, 27]. These works may suggest that in the event of failures, conversational speech might be able to help restore trust in the robot's capability, and that the content of the speech has an impact on the effectiveness of a robot.

Kollar et al. [24] extracted a sequence of directional commands from linguistic input for a humanoid robot or drone to follow. Tellex et al. [38] trained an inference model with a crowdsourced corpus of commands to allow humans to manipulate an autonomous robotic forklift with natural speech commands. In cases where the robot was told to perform an action that it did not understand, Cantrell et al. [7] demonstrated an algorithm where a human could explain the meaning of that action to the robot, and the robot would then be able to carry out instructions involving that action. Studies have also been conducted about how humans expect to interact with a robot [15, 17, 22, 23, 30]. There are also examples of robots that modify their speech behavior based on the circumstance and external conditions, such as [35, 40, 41].

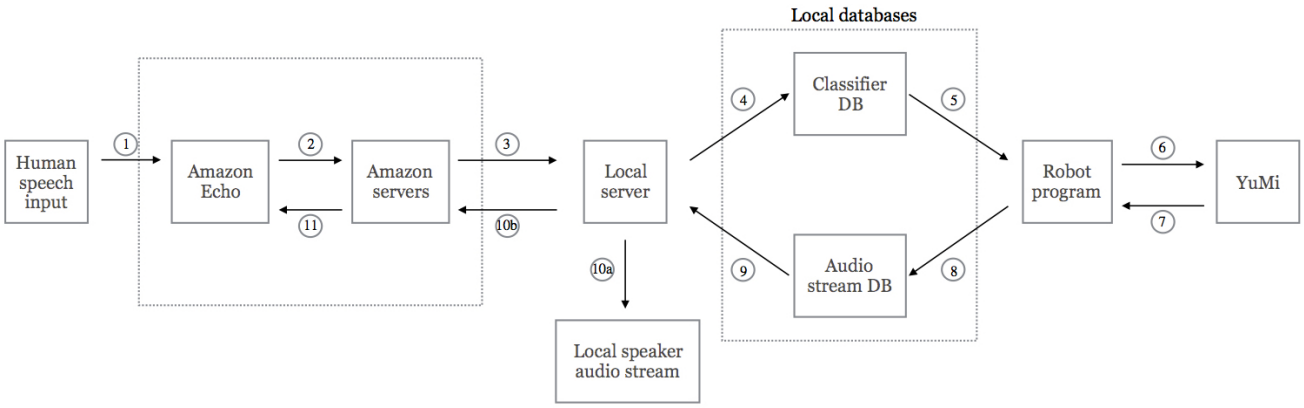

Fig. 2: System diagram. When (1) a user asks the Echo a question, (2) a request is sent to Amazon's servers over WiFi, which converts the speech to a robot command. (3) Amazon's servers send an HTTPS request to our server running on our local computer. (4) Our local server communicates actions to either the classifier or interaction database, depending on the command. (5) The robot manipulation program polls that database for a new command, and (6-7) communicates the actions and responses via Ethernet to the YuMi robot. (8) The robot program logs messages to the audio stream database, (9) which is polled by the local server. (10a) That message is played to the user through the computer's speakers in an audio stream. After the local server received the HTTPS request in step (2) and logs to the appropriate database (3), it returns an HTTPS response back to Amazon's server (10b), which relays the "end of request" command to the Echo (11).

III. System Design

A. System Architecture

To enable the Amazon Echo to communicate with the ABB YuMi robot, we implemented a web server on a local Linux desktop computer using the Django web framework, implemented in the Python programming language.

Our local Django web server is based on the `pycontribs/django-alexa` repository, which is publicly available on GitHub.com. We modified the code in the public repository to support the current format of Amazon's JSON messages and to handle our own custom application on the Amazon Echo. Our web application exposes a REST API endpoint at the relative address `/alexa/ask` of our server to communicate with the Amazon Echo (see Figure 2). This endpoint handles all incoming HTTPS requests and dispatches them to the appropriate Python functions that we define for various commands to EchoBot. We specify the public web address of our local server on the Alexa Skills Kit [3] web portal.

B. Communication with Amazon

Since we created a custom application on the Echo, Amazon requires that we specify a comprehensive, textual list of commands on the Alexa Skills Kit web portal beforehand. One or more invocation phrases, or human speech commands, must be specified for each robot command, and providing more phrases increases the robustness of Amazon's speech-to-command correspondence algorithm. Given this set of predefined possible human phrases to robot commands, Amazon's servers then compute the closest match of the speech to the corresponding command. These commands can contain parameters, or arguments, which are words or phrases that are variable in a given command. However, the set of possible parameters must be defined beforehand as well, which means that the system is unable to handle wildcard phrases for custom commands.

The Amazon voice service also places constraints on what a user must say to convey a human speech command. First, the Echo must be triggered by a "wake word", which can be either "Alexa", "Echo", or "Amazon", where we have chosen to use "Echo". Then, the user must specify their command in the form of <*action*> <*connecting word*> <*application name*> <*command*>, where we have named our application "YuMi", the connecting word is optional, and "command" refers to an invocation phrase. For example, to issue the command for the robot to grasp all parts, the user could say, "Echo, ask YuMi to pack all of the parts" or "Echo, tell YuMi to pack all of the parts". The command prefix must be included in the human speech command because we created a custom application with the Echo, rather than a native Amazon feature such as time or weather reports. If users want to issue frequent voice commands to the Echo, they may only say <*command*> for all following robot commands, provided that each command is issued within 5 seconds of the previous command.

When a user speaks a command to the Echo, an HTTPS request is sent over WiFi to Amazon's servers to convert the speech to text. The Amazon server then makes another HTTPS request in JSON format to our local server with the name of that command and potentially any parameters. The mechanics from the Amazon Echo to Amazon's servers are an abstraction, and the interface Amazon provides is speech as input and JSON data as output. Our local server parses the received JSON request and calls the function corresponding to that command name with any required or optional parameters. That function communicates the appropriate actions to a robot manipulation program via a database connected to our local server and accessible from anywhere on that same machine. The robot manipulation program communicates directly to the robot via Ethernet. Upon completion, the function on our local server sends an

HTTPS response, also in JSON format, back to Amazon's servers to be sent back to the Echo. This response may contain a phrase to be spoken through the Echo's speakers to the user, a signal for the Echo to continue listening for more commands, or a signal indicating the end of the command (see Figure 2).

The delay between the time the user finishes speaking to the Echo and the time our local server receives the HTTPS request is on average 2.1 seconds (see Figure 2, steps 1-3). The delay between the time the user finishes speaking to the Echo and the time the Echo receives the HTTPS response is 2.2 seconds (see Figure 2, steps 1-3,10b-11). This includes 0.5 seconds of delay after the user finishes speaking for the Echo to register that there is no more speech to send to Amazon's servers.

C. *EchoBot Audio Output*

The Amazon Echo API does not allow for the Echo to speak a series of phrases unless the user actively queries each one. Therefore, we routed the audio stream to the local server's computer speakers instead, to give the user explanation updates almost in real-time (see Figure 2).

To launch the audio stream, a user states, "Echo, ask YuMi to explain what it is doing." (See Figure 2 steps 1-3,10a.) Then, messages are logged from the main robot script to the audio stream database used by the local server (see Figure 2 step 8). Each record entry, or log, in the audio stream database consists of a time stamp and a logged message. As a message arrives into the audio stream database queue, the server converts the message into an audio signal using a text-to-speech library based on Google Translate's web API [39]. The server polls the database every 0.04 seconds, so the user is presented explanations with imperceptible latency as the messages arrive (see Figure 2 step 9). In the case where several messages are logged at the same time, only the last one is played, to reduce queuing delays. We have found that polling the database scales well as the total number of messages grows, since only the last message in the database needs to be checked at each poll.

Google Translate provides speech audio that sounds very natural, but because the text-to-speech library makes an HTTP request to convert the audio, translations would incur an average latency of approximately 2.5 seconds. Therefore, our local server caches these text-to-audio translations for immediate access, and defaults to a different text-to-speech library [31] that incurs latency on the order of 10 milliseconds.

We also utilized speech, music, and sound effects in an attempt to humanize the robot using EchoBot. When there are no explanations in the queue to play to the user, relaxing but interesting background music is played to fill in the silent gaps between messages. If the background music is not desired, there is also a command to disable it or play a different song using the built-in functionality of the Echo. With respect to data collection, we recognized that users may need detailed instructions while collecting initial data samples, but may quickly become annoyed at hearing complete details repeatedly, and might want shorter prompts as they become more comfortable with the task. EchoBot instead plays a short sound effect before each prompt for user action to condition the user to correlate the sound effect with the full message. Table I shows a sampling of the speech phrases that EchoBot uses to communicate with a user. Over successive sample trials, the speech message of the prompt reduces and is eventually discarded, leaving just the sound effect. The motivation for adding sound effects is to reduce the time it takes for EchoBot to relay instructions to the user, and improving the efficiency of data collection over speech instructions. The music and sound effects were added after our experiments to enhance the effectiveness of EchoBot based on our observations.

TABLE I: Example dialogues that EchoBot speaks to the user during data collection.

Sample Speech Output
Hello, friend! Let's get started.
Howdy, partner! Ready when you are.
Step 1: Please place the object in the workspace and capture an image.
Step 2: Please guide my arm to the object and record my arm's pose.
Step 2 is through, now run step 1.
Step 1 is done, now do step 2.
Looks like the gripper needs to be rotated 180 degrees.
Hmm, that's not quite right. Let's reset the sequence and try that again.
All done! You're really good at this!
Congratulations! You've finished the experiment.
Well done! You made it look easy!

D. *Data Collection using EchoBot*

Many LfD demonstrations require collecting poses or trajectories of the robot's arm positions as a human physically guides the arms and grippers. These trajectories may include several segments, where the endpoints of each segment are recorded. An example segment may be moving an arm's end effector from one location to another, or the actions of closing and opening the gripper. Robot arm trajectories can be recorded using buttons to specify the endpoints of each trajectory segment, where one button maps to a "start recording" command and another button maps to a "stop recording" command. This method has several shortcomings, including that demonstrators:

1) Often need both hands to control the robot's movements and can't stop to press a button.
2) Have to remember the mapping of buttons to commands.
3) Have to check the monitor display for cues on when the button press has been registered and the robot is prepared to record a trajectory.

EchoBot allows the user to speak commands such as "Start recording" and "Stop recording" to the Echo and achieve more intuitive control over data collection. Internally, when our server receives the user command, it sends the command to the main robot script via the classifier database (see Figure 2, step 4). Each record entry in the audio stream database consists of a time stamp, the logged command, and a Boolean

flag to indicate whether that command has been read. The classifier database is polled at least every 0.1 seconds by the robot program (see Figure 2, step 5).

IV. Human Performance Studies

A. *Grasp Task*

1) *Study Setting*

We evaluated EchoBot in a 2x2 study as an input and output interface to better understand the system's effectiveness in facilitating data collection for training a robot to grasp objects. The two input interfaces we compared were a keyboard button and EchoBot, and the two output interfaces were a monitor screen and EchoBot. The grasp task exemplifies a method to collect data for LfD robot learning. Given images of an object placed in various locations in the workspace and the poses of the robot to grasp those locations, LfD methods can be used to train the robot to grasp the object in unseen locations.

We used a factorial design to compare the four possible combinations of keyboard presses or EchoBot commands as input and a text-based monitor screen or EchoBot responses as output. Our subjects included 10 volunteers from our lab, who were randomly assigned to two of the four experimental conditions. Each condition had 5 subjects perform the experiment. For this task, we asked each subject to provide repeated grasp pose demonstrations on the YuMi robot kinesthetically. Each subject performed 10-minute experiments with two of the interfaces.

We measure the average durations of individual grasp trials and the percentage of failed grasps per condition.

2) *Study Procedure*

The experimenter provided each subject with the necessary information to perform the task using the I/O interface, including how to respond to potential errors that may occur. Subjects performed repeated trials for 10 minutes for each of two of the interface conditions. The order of the two conditions was randomized to mitigate learning effects. Subjects were given two minutes prior to each condition to be familiarized with the task and interface, to alleviate the effects of initial learning time.

Subjects were asked to provide input as a means to record demonstrations of correct poses for robotic grasps of the same object. They either stated "Echo, tell YuMi to record" for the EchoBot interface, or pressed the "r" key for the keyboard interface. By using the "r" key instead of the "Enter" key, the two input methods were more similar in terms of cognitive memory load on the user. The subject provides this input periodically every 20-30 seconds. The output messages of the monitor and EchoBot were exactly the same, with the function of giving the subject information about successes and failures while recording grasp poses. When the system detected a failure, such as if the gripper was outside the workspace, the subject had to re-demonstrate the grasp based on feedback from the output. After the subject completed both conditions, the subject was administered a short questionnaire, loosely based on [20], to understand user opinions of the interfaces.

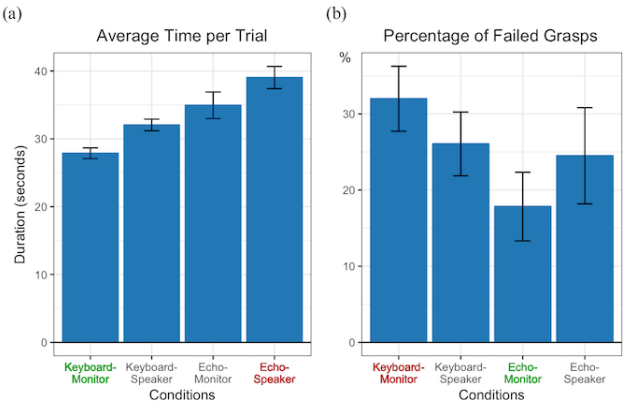

Fig. 3: Results for the grasp task, with 5 subjects per condition. The best condition is highlighted in green, and the worst in red. (a) Average durations of trials per condition. (b) Percent of failed grasps per condition in 10-minute tasks.

3) *Results*

To ascertain that initial learning effects did not cause subjects to increase in speed over time, we analyzed the durations of individual sample collections for each subject. We did not find a significant difference in these durations over time for any of the 4 interfaces.

We report the average durations of trials and number of failures per condition (see Figure 3), along with survey results. Each condition had between 74 and 110 total grasp trials across all participants. The average time per trial across conditions was found to be statistically significant, and the confidence intervals of all conditions were non-overlapping. Using the keyboard-monitor as the input-output interface has both more successes and failures, which means that this condition resulted in many more attempts at samples than the other conditions. However, survey results suggested that the Echo-speaker (EchoBot) and keyboard-speaker conditions were more intuitive and more enjoyable to use than the Echo-monitor and keyboard-monitor conditions, respectively. Although the most samples were collected with the keyboard-monitor interface, most users preferred to use an interface with either speech input or audio output.

The difference in successful grasps may be explained by the length of speech input and output in comparison with pressing a button or reading a sentence on the monitor. It takes a user longer to speak a 5-word command to the Echo than to press a key. Moreover, it was also observed that subjects would wait until EchoBot had finished its speech transmission before attempting the next trial, whereas they would immediately begin after the text on the monitor changed. The value of a speech interface may not have been apparent in this task, aside from user preferences, because the task did not require the user to engage both hands.

B. *Ring-Stacking Task*

1) *Study Setting*

Motivated by the findings of the grasping task, we designed a second task in which subjects provided full-trajectory, kinesthetic demonstrations of a ring stacking task with the YuMi robot. The task is similar to the Tower of

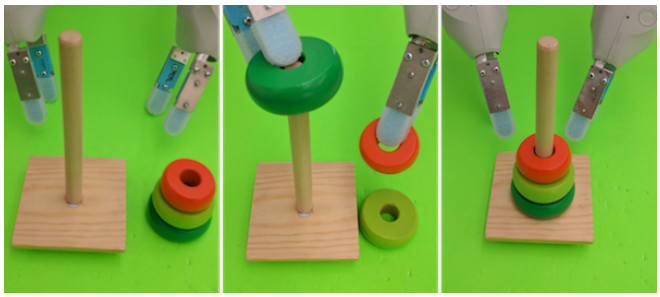

Fig. 4: The ring-stacking task (*left to right*). Subjects guide the robot grippers to pick up the rings to stack onto the rod in size order.

Hanoi puzzle, where the objective is to move 3 rings from a pile to a rod in size order using both robot grippers (see Figure 4). During the demonstrations, the user was asked to record every instance of the grippers being open or closed. Our aim was to compare the EchoBot interface with the keyboard-monitor interface for a task where the collection of human input was more frequent and occurred at inconsistent time intervals, and demonstrations required the concurrent involvement of both hands. While we did not use the data collected in the grasp task, this task is representative of other data collection methods that require constant input from a human. Frequent human input may be used in data collection for LfD methods in a dynamic workspace, and concurrent use of both hands allows for a larger range of tasks with the robot.

We used a within-subjects design, assigning 11 UC Berkeley computer science student volunteers to both conditions and randomly perturbing the order of the conditions. There was no overlap between participants in our two human performance studies. None of the subjects in this task had any prior experience with the Amazon Echo or with a mechanical robot. We measured the time to complete each demonstration and the percentage of human errors in record commands for each condition.

*2) **Study Procedure***

The experimenter provided each subject with the necessary information to perform the task using the I/O interface. The experimenter demonstrated a sequence to move the 3 rings from one pile to the rod and guided the subject to repeat the sequence twice prior to the experimental trials. This was to help the subject memorize the exact sequence of moves, and to reduce the effects of not knowing how to perform the task. During the trials, subjects repeated the same sequence and also indicated using the keyboard or EchoBot whenever either gripper opens or closes. This was done by saying "left opened", "left closed", "right opened", or "right closed" to EchoBot, or by typing "lo", "lc", "ro", or "rc" on the keyboard. If the subject made a mistake, the monitor or speaker informed the subject to restart the sequence before the trial was successful, yet continued to accept inputs. Thus, ignoring the output caused the subject to perform unnecessary work. After the subject completed both conditions, the subject was administered a short questionnaire, loosely based on [20], to understand user opinions of the interfaces.

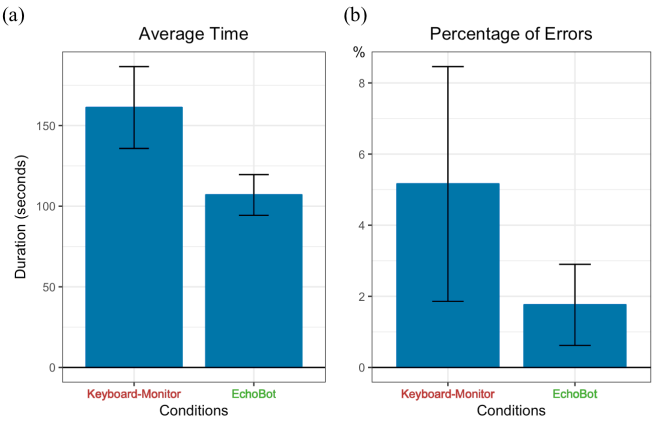

Fig. 5: Results from the ring-stacking experiment, with 11 trials per condition. The best condition is highlighted in green, and the worst in red. (a) Average durations of experiments across conditions. (b) Percentage of human errors across conditions.

*3) **Results***

Subjects using EchoBot were able to complete the ring-stacking task in less time than with the keyboard-monitor interface (see Figure 5). We observed a 57% decrease in average time to complete the task. All subjects were able to complete the ring-stacking demonstration with EchoBot in approximately equal or less time than with the keyboard-monitor interface. The difference between the average times per trial of the two conditions is statistically significant according to the Wilcoxon signed-rank test, and the confidence intervals of both conditions are non-overlapping. In addition, subjects committed fewer errors with EchoBot than with the keyboard-monitor. Survey results indicated that subjects found EchoBot to be more intuitive and enjoyable, and felt more efficient with EchoBot than with the keyboard-monitor.

Even though EchoBot outperformed the keyboard-monitor interface according to many metrics, there are still areas where it can be improved. Some subjects indicated that the 5-second listening timeout on the Amazon Echo was too short, and it was inconvenient to reactivate the Echo whenever they were unable to issue the next command within that time limit. EchoBot performed much better in this task than in the grasp task because this task required repeated input from the user every 5 seconds, and because it occupied both of the subject's hands. Frequent input, as opposed to every 20-30 seconds as in the previous experiment, allowed the user to reduce the size of the human speech command to the Echo by 4 words because the Echo can continue listening for input up to 5 seconds after a human speech command, a significant improvement for user interactivity. Moreover, we found that in the keyboard-monitor condition, subjects would often first speak the command (e.g. "left closed", "right opened"), think about which buttons to press, and then type the correct input, even if they had not yet experienced the EchoBot condition.

V. Discussion, Limitations, and Future Work

We present EchoBot, a system that uses the Amazon Echo to facilitate data collection for LfD. This paper presents an initial experimental study of the effects of using a voice

interface to collect data on a robot. The Echo is not robust to multiple voices speaking at once, and we anticipate that further shortening of phrase inputs will increase efficiency for data collection and that EchoBot can enhance the acceptance of robots.

VI. Acknowledgments

This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab, the Real-Time Intelligent Secure Execution (RISE) Lab, and the CITRIS "People and Robots" (CPAR) Initiative. The authors were supported in part by the U.S. National Science Foundation under NRI Award IIS-1227536: Multilateral Manipulation by Human-Robot Collaborative Systems and the Berkeley Deep Drive (BDD) Program, and by donations from Siemens, Google, Cisco, Autodesk, and IBM. We thank our colleagues who provided helpful feedback and suggestions, in particular Jeff Mahler, Michael Laskey, Sanjay Krishnan, Roy Fox, Daniel Seita, Nate Armstrong, and Christopher Powers.

References

[1] A. B. B. (ABB). (2017, January). [Online]. Available: http://new.abb.com/products/robotics/industrial-robots/yumi

[2] B. Akgun, M. Cakmak, J. W. Yoo, and A. L. Thomaz, "Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. ACM, 2012, pp. 391–398.

[3] Amazon. Alexa skills kit. [Online]. Available: https://developer.amazon.com/alexa-skills-kit

[4] ——. Amazon echo. [Online]. Available: https://www.amazon.com/echo

[5] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[6] J. M. Beer, A. Prakash, T. L. Mitzner, and W. A. Rogers, "Understanding robot acceptance," *Georgia Institute of Technology*, pp. 1–45, 2011.

[7] R. Cantrell, P. Schermerhorn, and M. Scheutz, "Learning actions from human-robot dialogues," in *RO-MAN, 2011 IEEE*. IEEE, 2011, pp. 125–130.

[8] E. Cha, A. D. Dragan, and S. S. Srinivasa, "Perceived robot capability," in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2015, pp. 541–548.

[9] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS quarterly*, pp. 319–340, 1989.

[10] L. A. Dean and H. S. Xu, "Voice control of da vinci," May 2011, voice integration with the da Vinci surgical robot, Johns Hopkins University. [Online]. Available: https://ciis.lcsr.jhu.edu/dokuwiki/lib/exe/fetch.php?media=courses:446:2011:446-2011-6:finalreport-team6.pdf

[11] M. Desai, P. Kaniarasu, M. Medvedev, A. Steinfeld, and H. Yanco, "Impact of robot failures and feedback on real-time trust," in *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2013, pp. 251–258.

[12] ——, "Impact of robot failures and feedback on real-time trust," in *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2013, pp. 251–258.

[13] M. Desai, M. Medvedev, M. Vázquez, S. McSheehy, S. Gadea-Omelchenko, C. Bruggeman, A. Steinfeld, and H. Yanco, "Effects of changing reliability on trust of robot systems," in *Human-Robot Interaction (HRI), 2012 7th ACM/IEEE International Conference on*. IEEE, 2012, pp. 73–80.

[14] M. Desai, K. Stubbs, A. Steinfeld, and H. Yanco, "Creating trustworthy robots: Lessons and inspirations from automated systems," *Proceedings of the AISB Convention: New Frontiers in Human-Robot Interaction*, 2009.

[15] B. R. Duffy, "Anthropomorphism and the social robot," *Robotics and autonomous systems*, vol. 42, no. 3, pp. 177–190, 2003.

[16] T. Fong, C. Thorpe, and C. Baur, "Collaboration, dialogue, human-robot interaction," in *Robotics Research*. Springer, 2003, pp. 255–266.

[17] S. R. Fussell, S. Kiesler, L. D. Setlock, and V. Yew, "How people anthropomorphize robots," in *Human-Robot Interaction (HRI), 2008 3rd ACM/IEEE International Conference on*. IEEE, 2008, pp. 145–152.

[18] L. Gallardo-Estrella and A. Poncela, "Human/robot interface for voice teleoperation of a robotic platform," in *International Work-Conference on Artificial Neural Networks*. Springer, 2011, pp. 240–247.

[19] Z. Henkel, V. Srinivasan, R. R. Murphy, V. Groom, and C. Nass, "A toolkit for exploring the role of voice in human-robot interaction," in *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2011, pp. 255–256.

[20] G. Hoffman, "Evaluating fluency in human-robot collaboration," in *International conference on human-robot interaction (HRI), workshop on human robot collaboration*, vol. 381, 2013, pp. 1–8.

[21] H. Jiang, Z. Han, P. Scucces, S. Robidoux, and Y. Sun, "Voice-activated environmental control system for persons with disabilities," in *Proceedings of the IEEE 26th Annual Northeast Bioengineering Conference (Cat. No.00CH37114)*, 2000, pp. 167–168.

[22] T. Kanda, T. Hirano, D. Eaton, and H. Ishiguro, "Interactive robots as social partners and peer tutors for children: A field trial," *Human-Computer Interaction*, vol. 19, no. 1, pp. 61–84, June 2004. [Online]. Available: http://dx.doi.org/10.1207/s15327051hci1901&2_4

[23] S. Kiesler and J. Goetz, "Mental models of robotic assistants," in *CHI'02 extended abstracts on Human Factors in Computing Systems*. ACM, 2002, pp. 576–577.

[24] T. Kollar, S. Tellex, D. Roy, and N. Roy, "Toward understanding natural language directions," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2010, pp. 259–266.

[25] M. Laskey, C. Chuck, J. Lee, J. Mahler, S. Krishnan, K. Jamieson, A. Dragan, and K. Goldberg, "Comparing human-centric and robot-centric sampling for robot deep learning from demonstrations," 10 2016. [Online]. Available: https://arxiv.org/abs/1610.00850

[26] C. E. Lathan and S. Malley, "Development of a new robotic interface for telerehabilitation," in *Proceedings of the 2001 EC/NSF Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly*, ser. WUAUC'01. New York, NY, USA: ACM, 2001, pp. 80–83. [Online]. Available: http://doi.acm.org/10.1145/564526.564548

[27] M. K. Lee, S. Kielser, J. Forlizzi, S. Srinivasa, and P. Rybski, "Gracefully mitigating breakdowns in robotic services," in *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2010, pp. 203–210.

[28] M. K. Lee and M. Makatchev, "How do people talk with a robot?: an analysis of human-robot dialogues in the real world," in *CHI'09 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2009, pp. 3769–3774.

[29] H. Medicherla and A. Sekmen, "Human-robot interaction via voice-controllable intelligent user interface," *Robotica*, vol. 25, no. 05, pp. 521–527, 2007.

[30] S. Nikolaidis, S. Nath, A. D. Procaccia, and S. Srinivasa, "Game-theoretic modeling of human adaptation in human-robot collaboration," *IEEE Transactions on Robotics*, January 2017. [Online]. Available: https://arxiv.org/abs/1701.07790

[31] P. Parente. Pyttsx. [Online]. Available: https://pypi.python.org/pypi/pyttsx

[32] C. Ray, F. Mondada, and R. Siegwart, "What do people expect from robots?" in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2008*. IEEE, 2008, pp. 3816–3821.

[33] B. Robins, K. Dautenhahn, R. Te Boekhorst, and A. Billard, "Effects of repeated exposure to a humanoid robot on children with autism," in *Designing a more inclusive world*. Springer, 2004, pp. 225–236.

[34] J. M. Sackier, C. Wooters, L. Jacobs, A. Halverson, D. Uecker, and Y. Wang, "Voice activation of a surgical robotic assistant," *The American Journal of Surgery*, vol. 174, no. 4, pp. 406–409, January 1997. [Online]. Available: http://dx.doi.org/10.1016/S0002-9610(97)00128-1

[35] D. Spiliotopoulos, I. Androutsopoulos, and C. D. Spyropoulos, "Human-robot interaction based on spoken natural language dialogue," in *Proceedings of the European workshop on service and humanoid robots*, 2001, pp. 25–27.

[36] S. S. Srinivasa, D. Ferguson, C. J. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. V. Weghe, "Herb: a home exploring robotic butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.

[37] L. Takayama, D. Dooley, and W. Ju, "Expressing thought: improving robot readability with animation principles," in *Proceedings of the 6th international conference on Human-robot interaction*. ACM, 2011, pp. 69–76.

[38] S. A. Tellex, T. F. Kollar, S. R. Dickerson, M. R. Walter, A. Banerjee, S. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation," *In Proceedings of the National Conference on Artificial Intelligence (AAAI 2011)*, 2011.

[39] Google, Inc. Google Text to Speech API. [Online]. Available: https://pypi.python.org/pypi/gTTS

[40] SoftBank Robotics. Pepper, the humanoid robot from Aldebaran, a genuine companion. [Online]. Available: https://www.ald.softbankrobotics.com/en/cool-robots/pepper

[41] A. R. Wagoner and E. T. Matson, "A robust human-robot communication system using natural language for harms," *Procedia Computer Science*, vol. 56, pp. 119–126, 2015.

[42] B. Wang, Z. Li, and N. Ding, "Speech control of a teleoperated mobile humanoid robot," in *2011 IEEE International Conference on Automation and Logistics (ICAL)*. IEEE, 2011, pp. 339–344.

[43] A. Weiss, J. Igelsbock, S. Calinon, A. Billard, and M. Tscheligi, "Teaching a humanoid: A user study on learning by demonstration with hoap-3," in *Robot and Human Interactive Communication (RO-MAN), 2009*. IEEE, 2009, pp. 147–152.