KRIS: A Novel Device for Kinesthetic Corrective Feedback during Robot Motion

Jorn Verheggen
Computer Science Department
Vrije Universiteit
Amsterdam, the Netherlands

Kim Baraka
Computer Science Department
Vrije Universiteit
Amsterdam, the Netherlands
k.baraka@vu.nl

***Abstract*—This paper presents a novel device that can be used to perform kinesthetic corrective feedback for robotic systems. KRIS (Kinesthetic Robotic Interaction System) is a device that can be mounted on the end-effector of an articulated robot. From here it can be manipulated by a human to give corrective feedback to the robot system during execution and in an intuitive way. The device can provide feedback in six degrees of freedom while giving passive haptic feedback to the user about both the position, rotation, and movement of the robot. We evaluated KRIS in a user study with respect to a baseline based on keyboard feedback in the areas of usability, intuitiveness, accuracy of corrections, and user task load. KRIS outperformed our baseline on the first three metrics and performed similar on task load. We believe that KRIS can enable a wide variety of robots to be taught interactively by non-expert humans in diverse collaborative settings.**

***Index Terms*—Learning from Demonstrations, Physical Human-Robot Interaction, Corrective Feedback, Robot Training Device, Human-Robot Collaboration**

I. Introduction

Most robotic systems today have at some point been explicitly programmed to perform a specific set of motions. This is often done by a technical expert, rather than someone with knowledge of the task or interaction at hand. While in some cases, domain knowledge is not crucial, as soon as there is an element of human-robot interaction this could become problematic. Robot motion that is explicitly programmed can often be inefficient [1], unintuitive, lacking legibility [2] and hence potentially unsafe. Most importantly, pre-programmed robot motions are difficult to adapt to new situations or to adjust to meet personalized end user needs and preferences.

This often leads to robotic systems being separated from humans. This separation occurs in terms of space (robots being fenced off from humans for safety or convenience) and in terms of collaboration dynamics (the robot performing its task after which the human performs theirs). We believe that by creating a system that democratizes the teaching of robotic motion we can bring robots and humans closer together and unlock a whole host of interesting possibilities for human-robot collaboration. We could for example create robotic systems that empower people to embed their task knowledge within the robot's behavior, or help robotic systems align with diverse user needs and preferences [3].

The field of human robot interaction (HRI) has gained considerable momentum over the past years. The literature now describes a large variety of topics relating to HRI including social navigation, tool transfer, preference learning, imitation learning, etc. With this variety of topics, a variety of devices that provide demonstrations or corrective feedback have been described as well. However, we believe that many of these input devices lack some of the basic characteristics that would allow non-expert users to program robot behavior in a flexible manner.

Bajcsy et al. (2017) claim that *"robots should treat physical human interaction as useful information about how they should be doing the task"* [4]. Along these lines, in this work we build on work in the field of Learning from Demonstrations (LfD) to inform the design of an interactive robot teaching system. The difference here will be that we will use these means of demonstration as means of corrective feedback *during* execution of robot motion. As a result, the user is able to refine the current robot motion through continuous interventions that interactively correct the current trajectory according to the user's kinesthetic input.

Many current input devices, such as joysticks or keyboards, are not designed to be friendly to non-expert users and are therefore not explicitly designed with intuitiveness in mind. The human often has to commit a chunk of their mental capacity to the mapping from interface inputs to robots outputs, or to observing the current behavior of the robot. This would take away from the mental capacity available for the task at hand in a collaborative setting. Other solutions, such as augmented reality or torque-based feedback, require expensive sensors or equipment and are hence not ideal. Our broad research question is therefore:
How can we facilitate kinesthetic teaching of autonomous robot motion in a manner that is usable, intuitive, and effective for non-expert teachers?

In answering this question we have created a device we call KRIS (Kinesthetic Robotic Interaction System). The design of KRIS makes it versatile with respect to a large variety of articulated robotic systems. The system contains an inner ring which is rigidly mounted on the robot (typically an end-effector), and an outer ring suspended by a series of springs around the inner ring. Thanks to magnetic sensors, this structure acts as a conduit for the human's kinesthetic input, in a way that is more loosely physically coupled than a system based on robot torque sensing [4]. Moving the outer ring with respect to the inner ring allows humans to give corrective feedback in all six degrees of freedom.

While holding on to the outer ring the humans also gets passive haptic feedback about the state and movement of the robot. Since this information is transmitted in a physical 3D environment close to the end-effector of the robot we hypothesize that most humans will be able to intuitively use the system as a teaching device in a dynamic and effective way. We expect this device to be useful in a variety of collaborative scenarios, for example in human-guided tool transfer [5] [6], or in personalizing existing robot trajectories for in-home robots (e.g., in cooking tasks).

The rest of the paper is structured as follows. In the next section we will review some of the HRI literature to identify desirable device requirements in relation to existing work. We will then present our design goals for KRIS, and describe its operation. We then evaluate our design by a series of experiments with a group of 12 participants. Lastly we discuss our findings, ending with a brief conclusion.

II. Related Work

Most work relevant to this paper comes from approaches and interfaces developed in the field of Learning from demonstration (LfD) [7] [8]. To this end, this work builds on insights from LfD to inform our decisions in creating a system for teaching autonomous robotic motion. The core distinguishing feature of this work is that it focuses on kinesthetic corrective feedback rather than offline demonstrations. To further inform our method, we first look at different modalities used to provide robot demonstrations, and then discuss methods used by the robot to interpret and integrate different types of corrective feedback (which can be treated as partial demonstrations).

A. Modalities of demonstrations

There broadly are three common categories for providing demonstrations to robots [8]. With *passive observation*, the robot passively observes an expert performing the task and attempts to imitate it through imitation learning methods. With *tele-operation*, demonstrations are given by means of a remote device like a keyboard or joystick. Finally, with *kinesthetic teaching*, the robot is physically moved to provide a demonstration. Passive observations are generally not applicable for our use case since they require actions to be performed sequentially rather then concurrently. However, the modalities of tele-operation and kinesthetic teaching are suitable for use with corrective feedback.

On the one hand, the effectiveness of *tele-operation* relies heavily on the type of input device used [9]. In the past people have used keyboards [10], joysticks [11], haptic interfaces [12], virtual or mixed reality interfaces [13] [14], etc. A common trait that these methods usually share is that they have low noise or bias. Where tele-operated demonstrations typically struggle is in the area of intuitiveness and ease-of-use. The user has to learn a mapping between their input and the device output, and have to train to effectively control a moving robot at a distance.

On the other hand, being able to quickly learn the mapping between user input and the resulting demonstration is where *kinesthetic teaching* excels. Prior research has shown that kinesthetic teaching can be performed by non-experts with minimal training [15] [16]. This is because this method requires direct manipulation of the robot's body, bypassing the need for learning a mapping from device controls to robot actions. A major downside of kinesthetic teaching is that it requires specialized hardware to be able to use it as corrective feedback. Most robot systems do not have precise internal sensors measuring the force applied to their joints. Another downside of traditional kinesthetic teaching is that manipulating a robot can be difficult, specially for large robots, or when demonstrations are trajectory-based rather than keyframe-based [15].

B. Incorporating demonstrations into learning

Once a demonstration has been obtained through some method of capture, the robot needs to process it in such a manner that it will learn the correct behavior. Learning low-level motion from human demonstrations or corrections is most commonly approached through a Reinforcement Learning (RL) formalism, whereby the robot learns a policy mapping a state space to a robot action space.

There are various algorithms aimed at achieving this goal. One such algorithm is COACH (COrrective Advice Communicated by Humans) [17]. COACH was initially conceived as a general learning algorithm but has later been expanded on to be applicable to robotic applications [18]. It works by incorporating a supervised learner module that supports the action selection. This supervised learner observes the state of the agent and its environment. It also takes in an advice signal from the human during the previous performed action. With this information the supervised learner produces a weight update from which the current policy is updated. This framework is then extended with the inclusion of a credit assigner module [19] and a human feedback modeler. Some agents can perform actions at such a high frequency that humans cannot adjust certain behaviors at the timescale required. The credit assigner module attempts to resolve this by associating the feedback not to a single action by the agent but rather to a window of previous actions.

The COACH algorithm has been extended by including a deep neural net to allow for larger input spaces [20]. This deep learning framework has later been enhanced to greatly decrease the time that is required to train an agent. It has also been adapted to include a forward dynamics model which makes it feasible to use in robotics applications. However, these additions of the COACH algorithm still use the credit assigner and the human feedback modeler that was introduced in the original COACH algorithm. We believe that we can largely eliminate the need for these modules by accounting for them in the design of our teaching device.

A device that has an analog output in all six degrees of freedom would eliminate some of the need for a human feedback modeler. The analog input could simply be directly related to the optimal policy. A device where the human moves concurrently with a robot could largely eliminate any lag between the robots actions and the human's corrections and would eliminate the need for a credit assigner.

III. KRIS DESIGN

In this section we first lay out our goals for the design of KRIS, then explain the working principle behind it.

A. Goals in designing KRIS

The design of KRIS took into account the following considerations. We first wanted a system that could provide analog corrective feedback to the robot in all six degrees of freedom (linear and rotational). We also wanted a just-right level of decoupling between the movement of the robot and the human. This would provide both some haptic pressure feedback to the human as to the direction of the robot's motion. Moreover, we wanted the system to be able to be mounted near the end-effector of the robot as to provide more precise feedback in the area of interest. And lastly we wanted the design to be adaptable to multiple articulated robot systems. With these requirement in mind we developed KRIS as described below.

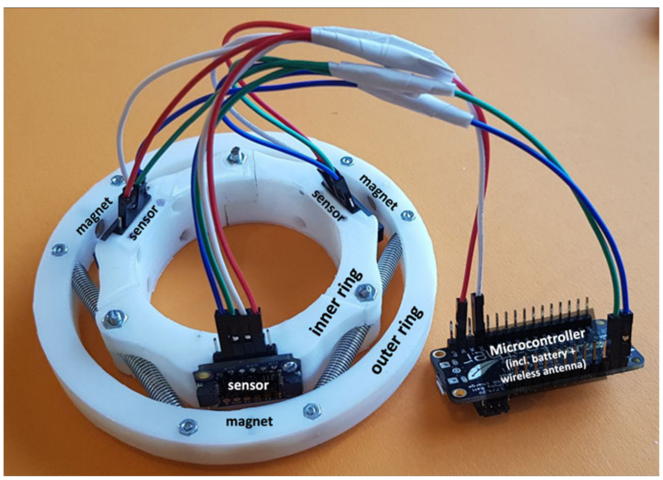


Fig. 1: Prototype of KRIS (Kinesthetic Robotic Interaction System).

B. Working principle

As shown in Fig. 1, the hardware of KRIS consists of two main sub-assemblies, the outer ring and the inner ring. The outer ring functions as a tactile interface for the human. It is suspended around the inner ring using an array of springs. This enables the outer ring to have a full six degrees of freedom whilst only being softly constrained. The inner ring functions as a reference point to the robot. It is rigidly attached to an extremity of the robot whose position is known through forward kinematics. Applying a force to the outer ring will change the relative position and rotation of the rings. The device is able to measure this discrepancy in position and rotation and use it as the user's indication of where they want the robot to move towards. Having an array of springs fundamentally alters the way of interacting with the robot compared to other solutions. It acts as a bi-directional interface for movement information. As the robot is moving the human is able to accurately feel how the rotation and translation of the robot is changing. They can then precisely intervene when it is required. On the side of the robot, the latter has access to continuous and real-time path correction information. Given Hook's law of spring constants, pushing with more force on the outer ring allows the user to specify by how much they would like to change the path of the robot. Lastly this method does not require any latency adjustment like other existing methods. In the case that the robot makes a sudden, unexpected and undesired movement the human will by simply holding on tight to the outer ring which will instantly send the right corrective feedback to the robot.

Some of the advantages over kinesthetic corrective feedback methods [4] [21] are that KRIS made of relatively cheap and off-the-shelf component. KRIS can be mounted on any existing articulated robot appendage as well as be integrated in new ones. KRIS also allows feedback to be given in any direction and any rotation at all times. This is sometimes not possible in kinesthetic methods due to issues like gimbal lock or other geometrical constraints.

C. Electromechanical design

For our prototype we have constructed the outer ring from two plastic 3D printed parts, namely a top and a bottom piece. It is clamped together using a set of nuts and bolts. Between the top and bottom piece three magnets are placed. These magnets are used as markers for specific points on the outer ring. The inner ring features three corresponding 3-axis magnetic sensors. These sensors measure the local magnetic field strength. Using this sensor information we get a 3D vector from the position of the sensor in the direction of the nearest magnet. The sensor information is then digitized and sent using the I2C protocol to a microcontroller. The microcontroller then sends this information wirelessly using UDP packets to an edge computer from which the robot's current trajectory is updated. The flow of information in the system is detailed in Fig. 2.

For our sensors we have chosen the MLX90393 sensor and a Adafruit breakout board. We specifically chose the MLX90393 for its very high dynamic range. Three-axis magnetic sensors tend to be very sensitive, which makes them pick up other magnetic radiation like the earth magnetic field and the magnetic induction of the robots actuators. This is additionally why we have relatively big magnets on the outer ring. We initially experimented with having smaller magnets in the hopes that the magnetic field would be pointing towards a smaller area but we achieved a better signal-to-noise ratio and improved accuracy by using bigger magnets. For the microcontroller we used the Adafruit Feather HUZZAH ESP8266 for the convenience of integrating a battery and wireless communication.

A repository including 3D models and an assembly guide for KRIS is made open-source and available at: https://github.com/kobotics/KRIS.

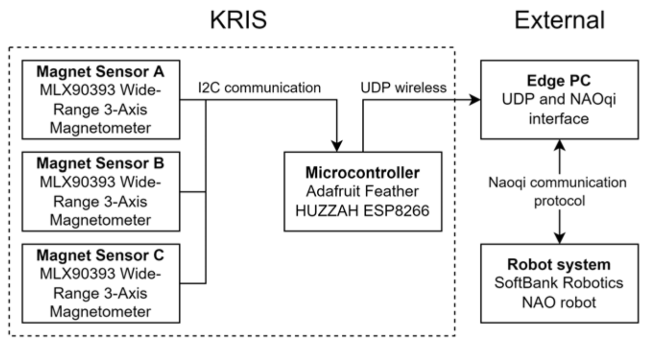


Fig. 2: System diagram of KRIS indicating the flow of information throughout the system. The bold text indicates the different electronic elements of the system and the non-bold caption indicates our specific choice for this prototype.

D. Transformation solver

The task of the transformation solver is to uncover the translation and rotation of the outer ring with respect to the inner ring. It does this by taking the measurements from each of the magnetic sensors (in the form of 3D vectors) and applying a series of mathematical operations to recover the homogeneous transform of the outer ring with respect to the inner ring. In order to do this efficiently we make use of three assumptions. (1) The magnetic flux radiates out radially from the magnet. Magnetic flux around a magnet in reality bends and warps all around the magnet, but since our sensors are placed relatively close to the magnet we can ignore this effect; (2) all magnets produce the exact same magnetic field strength; and (3) all sensors are identical. In reality there can be variations in how much some magnets are charged compared to others. This could cause aberrations in how the position of a magnet is measured. This position measurement could also be affected by differences in the individual magnet sensors. Different sensors could have different biases or different temperatures which could cause them to produce different values. In testing however, we found that these effects were small enough to be safely ignored.

Converting sensor readings into a six-dimensional corrective feedback vector involves the six following steps:

1) Expressing sensor readings in a common reference frame
2) Finding the mapping from sensor values to distance values
3) Calculating the positions of magnets from sensor readings through solving a constrained set of linear equations
4) Finding the transformation of the outer ring through the Kabsch algorithm [22]
5) Applying a soft maximum and minimum to avoid jerky motions at extremes
6) Applying a dead-zone to reduce sensitivity to noise due to external factors (see Section IV-A).

The last two steps were added as a result of pilot testing and user feedback.

IV. Evaluation

In order to evaluate the applicability of KRIS to kinesthetic teaching of robot motion, we selected four key success metrics to determine whether or not our design will be able to provide high quality corrective feedback. These are: (1) *usability* (during operation of the robot), (2) *intuitiveness*, (3) *accuracy of corrective feedback*, and (4) *task load*.

For each of our experiments we will detail how the experiment will impact these key success metrics.

A. Functional experiments

To test how well KRIS records the physical transformation of the outer ring we applied translations of the outer ring with various amounts and compared the transformation measured by KRIS with a measurement made with a pair of calipers, using the predicted magnet positions given by the transformation solver. This way we can witness individual aberrations between the sensors if they occur. Setup and results for linear axes are shown in Figs. 3 and 4. Tests on rotational axes are left out due to space constraints, but did not show major unexpected results.

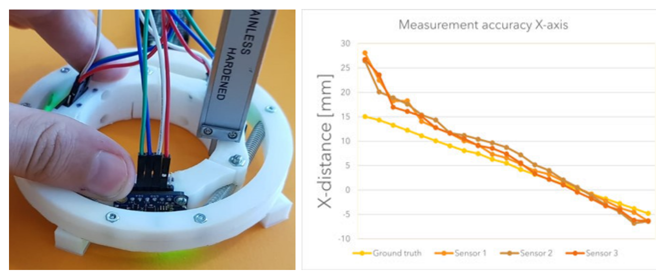


Fig. 3: X- axis accuracy setup and measurement

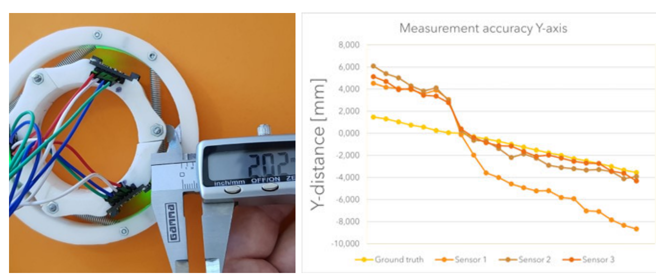


Fig. 4: Y- axis accuracy setup and measurement

From this test we see satisfactory performance in the X-axis direction. We do see the disparity between the caliper and measured distance increase above the 1cm range, however within this range the disparity is quite minimal and consistent. The Y-axis shows a bigger disparity between the calipers and the measured values, particularly in sensor 1, which was the sensor in-line with the translated axis. The measured values were still monotonically consistent with the calipers but show some nonlinearities. Since we average out the signals across the three sensors when computing the final homogeneous transformation, the signal is still usable but not as accurate as for the X-axis. The problem is likely that the sensors have a harder time determining the magnitude of the magnetic flux in comparison to direction. Possible solutions might be to change the layout of the magnetic sensors, or linearize the signal in a preprocessing step.

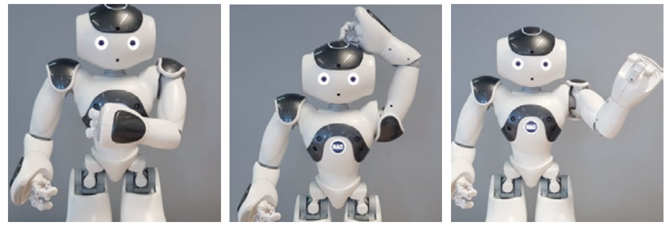
We also tested for potential outside influence related to factors such as the earth magnetic field, magnetic interference from the robot's actuators, oscillations, gravity, etc. through moving the robot to a pre-defined set of diverse positions over 10 trials. The average translation and rotation errors due to external factors was observed to be negligible (linear variation of a couple of tenths of a millimeter and rotational variation of +/- 0.02 radians), except for a significant aberration in the X-axis translation (up to 1.5mm). This is most likely due to the effects of the device's own weight when aligned with the horizontal plane.

B. User study

We conducted a user study with 12 participants (6 male and 6 female) in which we mounted KRIS on the arm of a NAO robot (see Fig. 5). After giving written informed consent, each participant was first given one minute to get familiar with the kinematics of the robot. This was done by allowing them to freely move the robot arm by hand. After this, the functionality of KRIS as well as the baseline was verbally explained. The participants were exposed to two conditions: *KRIS* and *keyboard*, each including one minute of familiarization. In the keyboard condition, each degree of freedom was mapped to a different key (4 arrows, right Shift, slash, 'q', 'w', 'e', 'a', 's', 'd'). In one of the two experiments, we also included a direct manipulation condition (see Section IV-B1). To factor out order effects, the order in which the conditions were presented was counterbalanced across participants. The study participants were then asked to participate in the Pose Mimic experiment explained in section IV-B1, then the Stop experiment explained in section IV-B2. Lastly the participants were asked to fill out a questionnaire (section IV-B3).

Fig. 5: User study setup. The left NAO is used to indicate the target pose in the Pose Mimic experiment, the middle NAO is used as baseline with the keyboard and the right NAO is mounted with KRIS.

1) Pose Mimic experiment: In the Pose Mimic experiment we took two static poses that were portrayed by a separate NAO robot. The task for our participants was to match the poses by using KRIS, and by using a keyboard. There was a time limit of 45 seconds imposed on this task. We first allowed the users to physically manipulate the robot (touch input) to determine what an ideal path from the base



(a) Starting pose (b) pose 1 (c) pose 2

Fig. 6: Mimic experiment poses

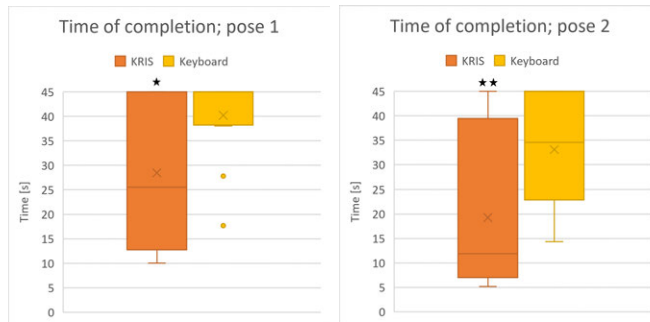


Fig. 7: Time of completion for both poses with both methods

position to the pose would look like according to the user. We chose to use a keyboard for our baseline to maximize the distinction in having a tactile based kinesthetic input method for KRIS as opposed to a more physically disjointed experience input method with the keyboard. During the experiment we recorded data about the path that the user makes with the robot arm, the inputs of KRIS, as well as the keyboard. This experiment is meant to test how well a user is able to direct the robot in the direction that they desire. For this experiment this is done without the robot moving in order to eliminate robot motion as a confounding variable. Figure 7 shows the time to completion of both poses. A short video snippet showing the KRIS condition in action is available at shorturl.at/lGTU8.

We observe that most users are able to complete the task faster with KRIS compared to the baseline. In total 15/24 trials were faster with KRIS with 3/24 being faster for the baseline and 6/24 being equal. A paired two sample t-test was performed to compare the completion time using KRIS and the keyboard. There was a statistically significant difference in completion time between KRIS ($M = 23.9$, $SD = 15.9$) and the keyboard ($M = 36.7$, $SD = 10.8$); $t(df) = -3.93$; $p < 0.001$). This indicates that KRIS performs statistically significantly better on the metric of completion time.

2) Stop experiment: The Stop experiment is aimed at investigating whether the user is able to give correct inputs to KRIS while the robot is moving. To this end, we let the robot move along a predefined path. Inputs given by the user in the direction of the path would make the robot speed up in. Conversely, inputs counter to the direction of the path would make the robot slow down. The task for the user was to slow down the robot as much as possible until ideally a complete stop. The slowdown was achieved by increasing the

Fig. 8: Scores of SUS and TLX section of questionaire.

interval time for the robot to reach the next position in its path. Users started out with an interval time of 0.02 seconds between path sections. If the user was able to slow the robot down to an interval time of 0.05 seconds the run would be considered successful.

The baseline generally outperformed KRIS in this instance with 8/12 successful runs for the baseline and just 3/12 successful runs for KRIS. However, while looking through the data we noticed that KRIS was in all cases able to significantly slow down the arm. However, the users struggled to find the correct direction to give feedback in while KRIS was moving slowly. If we for example take an interval time of 0.04 seconds, we notice that for 9/12 runs were successful and for KRIS 10/12 runs were successful. This indicates that if the arbitrary cutoff point was set to 0.04, KRIS would have narrowly outperformed the baseline.

In the future we would like to test KRIS on more meaningful tasks in combination with more complex plan update methods on the robot, potentially involving learning. We expect KRIS to be more usable in such contexts, especially in tasks with some level of tolerance on the optimality of the final policy.

3) Questionnaire: The questionnaire included two standardized scales to measure usability and cognitive load respectively, namely the the System Usability Scale (SUS) [23], and the Task Load Index (TLX) [24]. Results across both conditions are displayed in Fig. 8. A higher SUS score indicates the system is generally “more usable” and a higher TLX score indicates the system generally requires “more physical and cognitive load”.

A two sample t-test was performed to compare the SUS scores and TLX scores for using KRIS and the keyboard. There was a statistically significant difference in SUS scores between the keyboard ($M = 38.5$, $SD = 23.3$) and KRIS ($M = 62.1$, $SD = 12.5$); $t(df) = -2.63$, $p = 0.011$. There was no statistically significant difference in TLX scores between the keyboard ($M = 52.9$, $SD = 14.7$) and KRIS ($M = 46.7$, $SD = 11.4$; $t(df) = 1.68$, $p = 0.06$.

V. Discussion

In the area of usability, with the Stop experiment we demonstrated that KRIS is able to be effectively used during operation, and that it was deemed significantly more usable than our keyboard baseline. One caveat is that for lower robot speeds it became more difficult for the users to give the correct feedback to the robot. This is likely due to one of two factors: (1) the users found it difficult to judge how to correct the robot if it is moving slowly, or (2) the force that users exerted via the KRIS on the robot physically stopped it from moving, inhibiting them to see what the correct feedback would be. The former problem could be inhibited by researching other methods of obtaining the user’s desired policy. The latter problem could be solved by repeating the same experiment with a more powerful robotic platform (e.g., cobot arm).

In the area of intuitiveness, the system was tested giving participants minimal time to learn how to efficiently make use of the KRIS-robot system. With the Pose Mimic experiment we demonstrated that with this limited time users were still largely able to effectively give inputs in their desired direction. This shows that KRIS is generally more intuitive than the keyboard baseline.

In the area of accuracy, with both of our functional experiments we showed that the KRIS can accurately interpret user input. Furthermore we showed with both the Pose Mimic and stop experiments that the user is able to give precise feedback.

Lastly, in the area of task load, we found no statistically significant difference between KRIS and the baseline. This might be due to a discrepancy between the familiar nature / unfamiliar application of the keyboard interface and the unfamiliar nature / familiar application of the KRIS device. We expect task load to decrease for KRIS with repeated use of the device, and to stay stable for the keyboard condition (to be tested in future work).

VI. Conclusion

This work presented KRIS, a novel design for a device that enables end users to provide kinesthetic corrective feedback to robots in all six degrees of freedom (linear and rotational), during robot motion. Thanks to the spring system both the human and the robot have a level of independent movement. This also provides some haptic feedback to the human who is therefore more able to give effective corrective feedback. KRIS can be mounted on a large variety of articulated robotic systems. We have shown that the device is usable during operation of the robot, is intuitive, and offers precise corrective feedback without requiring a higher mental load than a keyboard baseline.

In future work, we would like to explore a more universal device design that can be mounted and calibrated on robots of varying sizes and shapes. We also would like to investigate how this type of corrective feedback is best interpreted by the robot as part of an interactive reinforcement learning algorithm. We view this work as a first step towards richer interfaces that empower human teachers by allowing them to more actively and efficiently adapt robot behavior to suit their needs in a variety of collaborative or assistive contexts.

VII. Acknowledgements

This work was developed as part of Jorn Verheggen’s Master thesis work within the Social AI lab at VU Amsterdam.

References

[1] K. Sohn and P. Oh, “Applying human motion capture to design energy-efficient trajectories for miniature humanoids,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3425–3431, 2012.

[2] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, “Legibility and predictability of robot motion,” in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 301–308, 2013.

[3] B. Woodworth, F. Ferrari, T. E. Zosa, and L. D. Riek, “Preference learning in assistive robotics: Observational repeated inverse reinforcement learning,” in *Proceedings of the 3rd Machine Learning for Healthcare Conference* (F. Doshi-Velez, J. Fackler, K. Jung, D. Kale, R. Ranganath, B. Wallace, and J. Wiens, eds.), vol. 85 of *Proceedings of Machine Learning Research*, pp. 420–439, PMLR, 17–18 Aug 2018.

[4] A. Bajcsy, D. Losey, M. O’Malley, and A. Dragan, “Learning robot objectives from physical human interaction,” 11 2017.

[5] T. Fitzgerald, E. Short, A. Goel, and A. Thomaz, “Human-guided trajectory adaptation for tool transfer,” in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS ’19, (Richland, SC), p. 1350–1358, International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[6] W. Li and M. Fritz, “Teaching robots the use of human tools from demonstration with non-dexterous end-effectors,” in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 547–553, 2015.

[7] 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.

[8] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, “Recent advances in robot learning from demonstration,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 297–330, 2020.

[9] W. Si, N. Wang, and C. Yang, “A review on manipulation skill acquisition through teleoperation-based learning from demonstration,” *Cognitive Computation and Systems*, vol. 3, no. 1, pp. 1–16, 2021.

[10] S. Osentoski, B. Pitzer, C. Crick, G. Jay, S. Dong, D. Grollman, H. B. Suay, and O. C. Jenkins, “Remote robotic laboratories for learning from demonstration,” *International Journal of Social Robotics*, vol. 4, pp. 449–461, Nov. 2012.

[11] C. Z. Qiao, M. Sakr, K. Muelling, and H. Admoni, “Learning from demonstration for real-time user goal prediction and shared assistive control,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3270–3275, 2021.

[12] Y. Liang, G. Du, C. Li, C. Chen, X. Wang, and P. X. Liu, “A gesture-based natural human-robot interaction interface with unrestricted force feedback,” *IEEE Transactions on Instrumentation and Measurement*, pp. 1–1, 2022.

[13] J. S. Dyrstad and J. R. Mathiassen, “Grasping virtual fish: A step towards robotic deep learning from demonstration in virtual reality,” in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1181–1187, 2017.

[14] S. P. Arunachalam, I. Güzey, S. Chintala, and L. Pinto, “Holo-dex: Teaching dexterity with immersive mixed reality,” *arXiv preprint arXiv:2210.06463*, 2022.

[15] 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*, HRI ’12, (New York, NY, USA), p. 391–398, Association for Computing Machinery, 2012.

[16] P. Kormushev, S. Calinon, and D. G. Caldwell, “Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input,” *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.

[17] C. Celemin and J. Ruiz-del Solar, “Coach: Learning continuous actions from corrective advice communicated by humans,” in *2015 International Conference on Advanced Robotics (ICAR)*, pp. 581–586, 2015.

[18] S. Jauhri, C. Celemin, and J. Kober, “Interactive imitation learning in state-space,” 08 2020.

[19] W. B. Knox and P. Stone, “Interactively shaping agents via human reinforcement: The tamer framework,” in *Proceedings of the Fifth International Conference on Knowledge Capture*, K-CAP ’09, (New York, NY, USA), p. 9–16, Association for Computing Machinery, 2009.

[20] R. Pérez-Dattari, C. Celemin, J. Ruiz-del Solar, and J. Kober, “Interactive learning with corrective feedback for policies based on deep neural networks,” in *Proceedings of the 2018 International Symposium on Experimental Robotics* (J. Xiao, T. Kröger, and O. Khatib, eds.), (Cham), pp. 353–363, Springer International Publishing, 2020.

[21] S. R. Ahmadzadeh, R. Kaushik, and S. Chernova, “Trajectory learning from demonstration with canal surfaces: A parameter-free approach,” in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pp. 544–549, 2016.

[22] W. Kabsch, “A solution for the best rotation to relate two sets of vectors,” *Acta Crystallographica Section A*, vol. 32, pp. 922–923, Sep 1976.

[23] J. Brooke, “Sus: A quick and dirty usability scale,” *Usability Eval. Ind.*, vol. 189, 11 1995.

[24] S. G. Hart, “Nasa task load index (tlx),” 2013.