Interactive Constrained Learning from Demonstration Using Visual Robot Behavior Counterfactuals

Carl Mueller
College of Engineering
and Applied Science
University of Colorado Boulder
Boulder, CO 80309-0422
Email: carl.mueller@colorado.edu

Aaquib Tabrez
College of Engineering
and Applied Science
University of Colorado Boulder
Boulder, CO 80309-0422
Email: mohd.tabrez@colorado.edu

Bradley Hayes
College of Engineering
and Applied Science
University of Colorado Boulder
Boulder, CO 80309-0422
Email: bradley.hayes@colorado.edu

***Abstract*—Collaborative robots continue to depend on substantial robot programming expertise to be useful to end-consumers and small to mid-level enterprise. Robot skill learning techniques, like Concept Constrained Learning from Demonstration, allow a robot to learn robust skills from non-expert users. This method combines traditional Robot Learning from Demonstration data with constraints to enable the communication of richer skill-pertinent information as task specific behavior restrictions. This approach is integrated into a visual interactive system called Augmented Reality for Constrained Learning from Demonstration (ARC-LfD). This interactive system enables users to iteratively program robot skills through demonstration and constraint application *in situ* using augmented reality.**

However, as constraints and acquired skills grow in number, users might not have a deep understanding of the capabilities of the robot for any given learned skill. This paper proposes an extension to the ARC-LfD system that will provide 'what-if' visualizations called *Robot Behavior Counterfactuals* (RBCs). RBCs serve to explain the effects of alternative constraint usage, as well as the effects constraints have on the potential for skill success, particularly when adapting skills to altered environments. ARC-LfD will also be extended with visuals called *Behavioral Verification Indicators* that aid users in understanding where and why a potential model will fail or succeed. This proposed system will be evaluated with a human-trial study to test for objective and subjective measures of belief in robot capability.

I. Introduction

Collaborative robots (cobots) are those designed to work with or alongside human counterparts. They have the potential to greatly expand physical automation in small to mid-level enterprises as well as in end-consumer applications. However, cobots remain inaccessible to the vast majority of end-users due, in part, to the extensive robot programming knowledge needed for successful deployment. This is particularly true due to the nature of the tasks cobots are expected to perform within human-robot settings: safety in shared environments, dynamic task requirements, decision-making, and adhering to user expectations of behavior. One key in overcoming these challenges is to inject an element of intelligent self-autonomy and awareness that boost a robot's intrinsic capability.

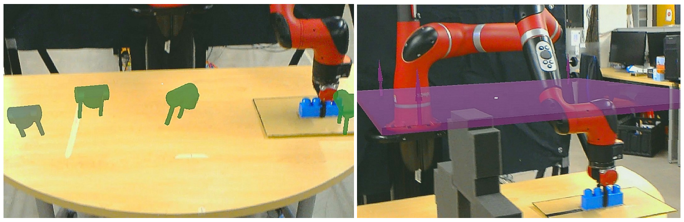

Fig. 1. Example of ARC-LfD visualizations with an expected trajectory (left) and possible constraint (right) as augmented reality visuals. Robot Behavior Counterfactuals will integrate into these visuals as multiple alternative execution trajectories overlaid within the same point of view, offering users a preview of potential alternative execution capabilities.

A set of learning techniques that employ statistical and machine learning methods is called Robot Learning From Demonstration (LfD). Inspired by human-to-human teaching, the goal of LfD methods is to enable non-expert user to teach robots skills without extensive programming knowledge. By leveraging the domain-knowledge of the user, LfD techniques have users 'demonstrate' to the robot how the user wants the robot to complete a task. Ideally, the robot possesses the appropriate intelligent learning ability such that it captures enough information from the user to produce a successful skill model.

One LfD method called *Concept Constrained Learning from Demonstration* (CC-LfD) augments traditional 'demonstration' data (e.g. robot configuration trajectories) with high-level behavioral restrictions or constraints [23]. Along with physically moving the robot through the skill it should learn (i.e. demonstration), users also demarcate when and where behavioral restrictions must be satisfied during execution. This algorithm has been integrated into a system called *Augmented Reality for Constrained Learning from Demonstration* (ARC-LfD) that produces visualizations of expected robot execution trajectories and constraints [20] (see Fig. 1). ARC-LfD enables non-expert users to train robot skills using CC-LfD through demonstration and interactive constraint engineering to generate new learned models. This facilitates skill model shaping to adapt to new task requirements such as changes to goal/target orientations and new arrangements of the environment.

However, as the number of constraints and acquired skills are made available to the robot increases, users might not have a clear understanding of the effects of constraints or even if the chosen learned model can adapt to changes in the environment. This paper proposes that extending ARC-LfD with concurrent visualizations of alternative execution models called *Robot Behavior Counterfactuals* (RBCs) and additional success/failure visuals called *Behavior Verification Indicators* (BVIs) will increase awareness of learned model capability by illuminating the effects of changing constraints and by explaining the ability of a model to adapt to new task requirements.

II. Related Works and Preliminaries

A. Constrained Learning from Demonstration

Robot Learning from Demonstration (LfD) methods are those that enable the learning of successful robot behavior models from human demonstration [4]. Through various modes of interaction, a human operator provides a set of demonstrations that ideally communicate the nature of the skill to the robot system. The goal is such that the expertise of the human translates to a more successful learned model than more brute force techniques such as reinforcement learning-based models [3]. Most importantly, LfD methods reduce the need for robotics expertise, enabling non-roboticists to quickly teach robots useful skills.

The modes of demonstration traditionally employed by LfD methods generally focus on the kinematic aspects of the demonstrated skill, be it imitation-based or kinesthetic-based demonstration [5, 8, 1]. In other words, very often the demonstration focuses on the physical movements required to execute the task successfully. However, if we look to human-human teaching for inspiration, we see there are other information-dense forms of communication (e.g., speech and gesture) as well rich implicit contexts highly beneficial to knowledge transfer [24, 36]. Mueller et.al. [23] introduces the idea of incorporating conceptually-grounded behavioral constraints to enrich the information usable by Keyframe-based Learning from Demonstration techniques [2] in an algorithm called Concept Constrained Learning from Demonstration (CC-LfD). Concept Constraints are semantically grounded Boolean classifiers that determine whether or not an environment state vector satisfies the represented constraint (e.g. keeping a cup upright, or maintaining a specific distance from an object). These constraints are communicated to the robot system by the user, usually during demonstration. Concept Constraints help shape the learned model to more closely reflect the ground truth representation of a skill by encoding richer and more abstract information than can be done by traditional demonstration alone.

An important caveat is that CC-LfD in its current form occludes the effects of newly assigned constraints, only revealing the relearned model during execution. As constraints and their effects on robot behavior might not always be comprehensible to the human operator, this is potentially problematic. As these Boolean classifiers grow in complexity (such as an increase in the number of parameters), it might be difficult for the human operator to intuit whether such constraints are useful or what their effect might be on the behavior of the robot. Conversely, a user might be quite aware of the behavioral restrictions needed to be placed onto a robot but might not know whether doing so is feasible given the current learned model or environment.

With these limitations in mind, an alternative interface might better inform the user of the potential effects of constraints. One such interface is Augmented Reality (AR), which has a well-established body of research in the robotics community [12, 35] as evidenced by improving human-robot teaming [9, 29], increasing safety in shared environments [40, 30], and through interfaces for explainability [16, 17, 11]. Motivated by this research, Luebbers et.al. [20] developed an interface the combines the benefits of the enriched information communication of CC-LfD with AR. This system is called Augmented Reality for Constrained Learning from Demonstration (ARC-LfD) and it enables users to interactively update constraints and observe the corresponding relearned models given the new restrictions on the behavior of the robot. ARC-LfD provides visualizations of both constraints assigned to segments of a skill, and the expected trajectory of execution the robot will undergo (see Fig. 1). The motive is that this knowledge will better communicate to the user whether or not the robot will adhere to notions of correct behavior and whether it will successfully execute the task.

B. Counterfactuals in Robotics

CC-LfD and ARC-LfD constitute a novel extension in the field of LfD, but they are limited in that they require the user to internally think about what constraints might be needed to correctly shape a learned model. While ARC-LfD can provide model visualizations, it does not currently support a way to *compare* differences in constraint assignment and parameterization. Ideally, this system should enable users to ask "*what if?*" questions that answer how different constraint assignments and parameterizations will look compared against each other overlaid in the same visualization. Conversely, the system could show the user what would happen if a constraint were suddenly removed from the learned model. These approaches of generating hypothetical alternatives given changes to the learned model's expected execution can be considered a form of a *counterfactual*.

Generally, contrastive examples and counterfactuals are approaches for intuiting the validity of causal models. Contrastive examples are those that offer an alternative scenario in order to exemplify the validity of the current model or to intuit input/output relationships [19]. Contrastive examples alter inputs to a known model to determine the role such changes have on the model's output. This essentially seeks to identify which input features play a role in affecting output thereby providing an accounting for which inputs or features have a causal effect. Counterfactuals [22] use negating hypothetical causal conditionals to intuit the causal link between one event and another. For example, given the statement, “I drank hot tea, and my mouth is burned”, a counterfactual explanation would be, “If I did *not* drink hot tea, then I wouldn’t have burned my mouth”. In other words, a counterfactual is a hypothetical causal statement that describes an alternative scenario that investigates whether the negation of one event demonstrates a causal link to the other event.

Contrastive explanations and counterfactuals are well utilized within Artificial Intelligence (AI) [25, 10], but are especially leveraged within the subfield of Explainable AI (XAI). These techniques are useful for generating explanations via natural language [13], for enhancing explainability in model-free reinforcement learning [21], for use alongside a game-theoretic framework to generate machine learning model explanations [28], and for debugging machine learning models [39]. In robotics, counterfactuals have been used to demonstrate causal reasoning about tool affordances [7], as a mechanism to reason about robot navigation [6], and as a mechanism for causal inference and explanation to enhance robot control [33].

Counterfactuals are traditionally used as a mechanism to verify a known causal structured model [27], but in machine learning such causal models do not exist or are opaque to the user. Therefore, counterfactuals generally become a mechanism for intuiting the effects of varying input on the model’s output, but not necessarily for intuiting the causal mechanism for that change [26]. Generally, counterfactuals in AI/ML are used as a reasoning / debugging aid for human understanding but not as a mechanisms that the model itself uses to reason about causality [39]. The proposed Robot Behavior Counterfactuals are not used by robot learning system to reason but instead to allow human operators to reason about the validity of the model they are teaching the robot.

III. Robot Behavior Counterfactuals

A. RBC and BVI Visualizations

We posit that presenting counterfactual behavior visualizations of the effects constraint assignment and parameterization will enhance the ARC-LfD interface by providing possible alternative execution models that better informs users about model capability and the necessity of constraints. Figure 2 acts as a simplified example of this concept. In their simplest form, RBCs are models (and the corresponding visualizations) of expected robot behavior given user updates to a set of assigned constrained on a CC-LfD model using the ARC-LfD interface. RBC constraint (Fig. 2, purple) and trajectory (Fig. 2, orange) visualizations will be placed in the same augmented reality environment as the current expected trajectory visualization (Fig. 2, black). We also posit that visual indicators that inform users of potential skill execution success or failure will also enhance the ARC-LfD interface (Fig. 2, red & green). These Behavior Verification Indicators will provide visual aids to users that indicate whether or not changes to a model will result in a success or failure that might not easily be captured by a planning predicate in the form of a Concept Constraint.

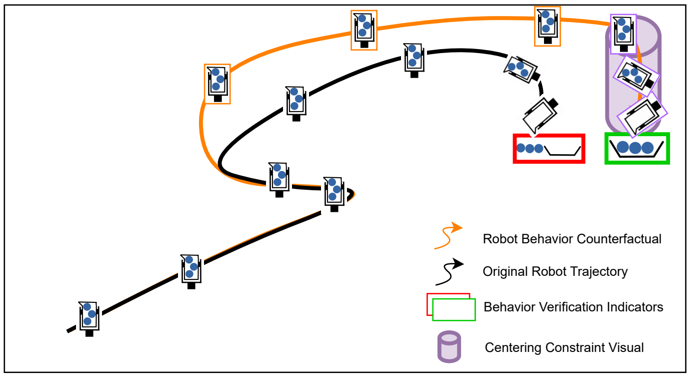
Robot Behavior Counterfactual
Original Robot Trajectory
Behavior Verification Indicators
Centering Constraint Visual

Fig. 2. Simplified visual of a Robot Behavior Counterfactual (orange) overlaid in the same field of view of the current expected robot trajectory (black). An example Behavior Verification Indicator might highlight how the skill failed (red) or succeeded (green). The RBC differs from the original trajectory due to the assigned centering constraint (purple) that makes the robot pour the contents correctly into the receptacle.

Figure 3 provides a high-level overview of the proposed interaction flow of ARC-LfD extended with RBCs and BVIs. 1) A user first provides a set of initial demonstrations along with a potential set of constraints to produce an initial CC-LfD model. 2) The ARC-LfD visualization engine takes this initial model and provides a visualization of the expected robot behavior. 3) The user is given the option to add/remove/update constraints to generate potential parameterizations for alternative models. 4) These are fed into an update process where multiple models are generated, including the potential evaluation of the success/failure modes of those models. These generated models serve as the basis for the RBCs visuals produced again by the ARC-LfD visualization engine. Steps 2-4 can be repeated until the user elects to choose the model of the RBC that is ideal for actual execution, step 5.

B. Technical Extensions to ARC-LfD

In order for ARC-LfD to provide RBC visualizations, the system must be extended in two important ways. The first need is for the interface to display multiple constrained trajectory visuals overlaid in the same point of view of the Microsoft Hololens. This requires the ability to run multiple concurrent model updates and the ability to quickly produce constrained motion plans. ARC-LfD only produces visualizations of the sequences of the keyframe waypoints of these models. With the addition of visualized trajectories of the end-effector, we can provide animations (leveraging simulation) between these keyframes. This will provide a visual example of how the robot will execute a skill, adhere to assigned constraints, and interact with its environment. In order to produce simulated animations, the system will also need to quickly produce feasible constrained motion plans for the variations of RBC’s dependent on differing constraint parameterizations and assignments.

The second extension to ARC-LfD will be the addition of Behavior Verification Indicators (BVIs) that aid in the use of the visual explanations for skill success/failure. BVIs will be

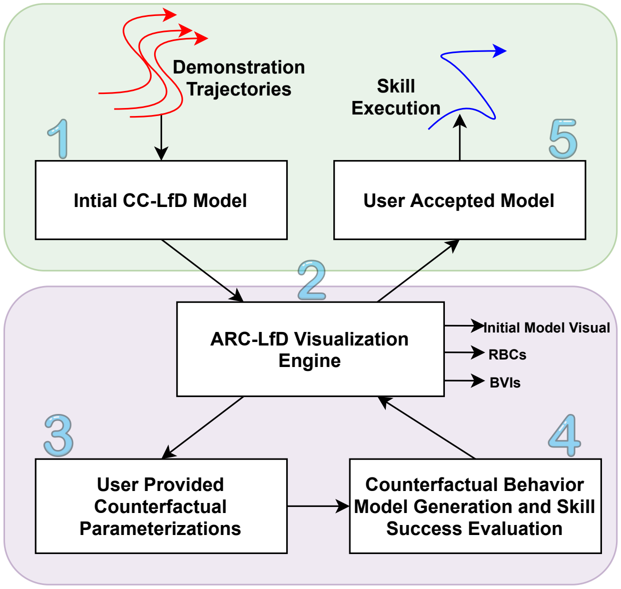
Demonstration Trajectories
Skill Execution
1
5
Intial CC-LfD Model
User Accepted Model
2
ARC-LfD Visualization Engine
Initial Model Visual
RBCs
BVIs
3
4
User Provided Counterfactual Parameterizations
Counterfactual Behavior Model Generation and Skill Success Evaluation

Fig. 3. High-level overview of the interaction flow of ARC-LfD now extended with Robot Behavior Counterfactual and Behavior Verification Indicator visualization. Green zone indicates physical interaction. Purple zone indicates interaction in AR.

visually similar to how ARC-LfD currently displays constraint satisfaction given expected robot behavior, but they do not result in model relearning. For example, a BVI might indicate whether or not contents from a cup successfully reached a target receptacle. This indication could take the form of a green box outline around the target if the simulation of an RBC trajectory results in the contents successfully placed into the receptacle (see Fig. 2). Such an indication is not easily grounded into a Concept Constraint Boolean classifier for use by the underlying CC-LfD algorithm. Not all indicators of success can be used as planning predicates.

IV. Study Design and Evaluation

A. Interface Design Scenarios

The purpose of this proposed evaluation is to demonstrate that the information provided by ARC-LfD, and especially the extensions of ARC-LfD that support RBCs, will increase the situational awareness of human users of the system. Specifically we will conduct four different conditions as a between-subjects study. Each condition is designed to build up from a baseline that mimics the original CC-LfD evaluation, tests the current ARC-LfD system, and then evaluates the addition of RBC's with and without BVIs. For each condition, given previously learned skill models, the user must decide whether or not the resultant updates to constraints will result in skill execution success. A between-subjects study design is chose to minimize learning effects between the differing interface capabilities.

1) *Condition I - Baseline:* In the control condition, we show the users the demonstration of kinesthetic teaching by adding constraints through a tablet interface, and then demonstrate the resulting execution behavior as the only information update about the effects of constraints. There is no ARC-LfD interaction in this condition as it mimics the original evaluation of CC-LfD.
2) *Condition II - Current ARC-LfD Visualizations:* Uses the existing ARC-LfD interface without RBCs where that only shows updated models after constraint updates are performed.
3) *Condition III - ARC-LfD with RBC:* In this condition, side-by-side visualizations of models generated from proposed constraint changes (RBCs) are provided to the user, who can either accept or reject the differing RBCs.
4) *Condition IV - ARC-LfD with RBCs and BVIs:* This condition adds BVIs to Condition III to further aid the users in knowing whether a task is successfully executed or not.

B. Experiment Tasks

We will be using same three task presented in the case studies of ARC-LfD for consistency for each condition:

1) *Pouring/Transferring Task:* The robot should carry a cup filled with contents, move the arm over an obstacle, position the cup over a target, and pour out the contents into a target bowl.
2) *Lid Placement Task:* The robot should place a lid correctly onto the center of an object top. Two obstacles on either side of the target will require the end-effector to smoothly travel between the objects without knocking them over.
3) *Cubby Task:* The robot should place an object inside a cubby that requires proper orientation of the object for successful placement.

C. Design Validation

For each conditional, participants will complete surveys before and after task execution to test the following objective and subjective metrics:

1) *Trust/Reliability:* We will use established scales in HRI (HRI Trust Scale, Dyadic Trust Scale) to determine the subjective trust and reliability of the different design interfaces [18, 32].
2) *Helpful/Useful:* Helpfulness and usefulness scales provide the subjective measure of usefulness of the system to the user [37, 34].
3) *Explainability/Interpretability:* Explainability and interpretability metrics try to capture users' mental models of intelligent systems or decision aid systems. We will be using explainability metrics to determine users' understanding and trust in the system [15, 38].
4) *Shared perception and situational awareness:* Shared perception or shared mental model allows teams or collaborators to draw from the common knowledge to perform fluent and adaptive actions increasing trust and teamwork [38]. Situational awareness defines the informational needs of humans during any activity or

collaboration. Shared perception and situational awareness are correlated with increased explainability and improved team performance [31].

As per the subjective measure's requirement, we will be using established practices in HRI for operationalizing constructs into variables and measures as provided by Hoffman and Zhao [14].

We hypothesize the following: H1) conditions 2-4 will generally outperform conditions 1 for all the given metrics, H2) condition 3 and condition 4 will provide more explainability, trust, and reliability compared to other conditions because of the increase in conveyed information provided by RBCs and BVIs, and H3) condition 4 will also outperform other scenarios in terms of trust/reliability and usefulness, as the BVIs aid users to provide closure on whether a task is successfully learned.

V. Conclusion

In this work, we propose an extension of the constrained LfD interface ARC-LfD by allowing human operators to reason about the validity of the model learned by the system. We leverage counterfactual explanations from explainable AI literature to generate *'what-if'* visualizations called Robot Behavior Counterfactuals. Additionally, we propose Behavioral Verification Indicators to aid users in understanding why a model fails or succeeds. We also outline a human-subject study to evaluate our proposed extension.

Acknowledgments

This work was supported in part by NSF NRI Award 1830686 and ARL STRONG Cycle 2.

References

[1] B. Akgun, M. Cakmak, J. W. Yoo, and A. L. Thomaz. Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective. In *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 391–398, March 2012. doi: 10.1145/2157689.2157815.

[2] Baris Akgun, Maya Cakmak, Karl Jiang, and Andrea L Thomaz. Keyframe-based learning from demonstration. *International Journal of Social Robotics*, 4(4):343–355, 2012.

[3] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57 (5):469–483, 2009.

[4] Christopher G Atkeson and Stefan Schaal. Robot learning from demonstration. Citeseer.

[5] Paul Bakker and Yasuo Kuniyoshi. Robot see, robot do: An overview of robot imitation. In *AISB96 Workshop on Learning in Robots and Animals*, pages 3–11, 1996.

[6] Alejandro Bordallo, Fabio Previtali, Nantas Nardelli, and Subramanian Ramamoorthy. Counterfactual reasoning about intent for interactive navigation in dynamic environments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2943–2950. IEEE, 2015.

[7] Jake Brawer, Meiying Qin, and Brian Scassellati. A causal approach to tool affordance learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8394–8399. IEEE, 2020.

[8] Sylvain Calinon and Aude Billard. Incremental learning of gestures by imitation in a humanoid robot. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, pages 255–262. ACM, 2007.

[9] Kishan Chandan, Vidisha Kudalkar, Xiang Li, and Shiqi Zhang. Negotiation-based human-robot collaboration via augmented reality. *2019 AAAI Fall Symposium: AI for HRI*, 2019.

[10] Mark Derthick. Counterfactual reasoning with direct models. In *AAAI*, pages 346–351, 1987.

[11] Maximilian Diehl, Alexander Plopski, Hirokazu Kato, and Karinne Ramirez-Amaro. Augmented reality interface to verify robot learning. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 378–383. IEEE.

[12] Scott A Green, Mark Billinghurst, XiaoQi Chen, and J Geoffrey Chase. Human-robot collaboration: A literature review and augmented reality approach in design. *International Journal of Advanced Robotic Systems*, 5(1): 1, 2008.

[13] Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. Generating counterfactual explanations with natural language. *arXiv preprint arXiv:1806.09809*, 2018.

[14] Guy Hoffman and Xuan Zhao. A primer for conducting experiments in human–robot interaction. *ACM Transactions on Human-Robot Interaction (THRI)*, 10(1):1–31, 2020.

[15] Robert R Hoffman, Shane T Mueller, Gary Klein, and Jordan Litman. Metrics for explainable ai: Challenges and prospects. *arXiv preprint arXiv:1812.04608*, 2018.

[16] Kazuhiko Kobayashi, Koichi Nishiwaki, Shinji Uchiyama, Hiroyuki Yamamoto, Satoshi Kagami, and Takeo Kanade. Overlay what humanoid robot perceives and thinks to the real-world by mixed reality system. In *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 275–276. IEEE, 2007.

[17] Tomáš Kot and Petr Novák. Utilization of the oculus rift hmd in mobile robot teleoperation. In *Applied Mechanics and Materials*, volume 555, pages 199–208. Trans Tech Publ, 2014.

[18] Michael Lewis, Katia Sycara, and Phillip Walker. The role of trust in human-robot interaction. In *Foundations of trusted autonomy*, pages 135–159. Springer, Cham, 2018.

[19] Peter Lipton. Contrastive explanation. *Royal Institute of Philosophy Supplements*, 27:247–266, 1990.

[20] Matthew B Luebbers, Connor Brooks, Carl L Mueller, Daniel Szafir, and Bradley Hayes. Arc-lfd: Using aug-

mented reality for interactive long-term robot skill maintenance via constrained learning from demonstration. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[21] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Distal explanations for model-free explainable reinforcement learning. *arXiv preprint arXiv:2001.10284*, 2020.

[22] Stephen L Morgan and Christopher Winship. *Counterfactuals and causal inference*. Cambridge University Press, 2015.

[23] Carl Mueller, Jeff Venicx, and Bradley Hayes. Robust robot learning from demonstration and skill repair using conceptual constraints. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6029–6036. IEEE, 2018.

[24] Chrystopher L Nehaniv and Kerstin Ed Dautenhahn. *Imitation and social learning in robots, humans and animals: behavioural, social and communicative dimensions.* Cambridge University Press, 2007.

[25] Charles L Ortiz Jr. Explanatory update theory: Applications of counterfactual reasoning to causation. *Artificial Intelligence*, 108(1-2):125–178, 1999.

[26] Judea Pearl. The seven tools of causal inference, with reflections on machine learning. *Communications of the ACM*, 62(3):54–60, 2019.

[27] Judea Pearl et al. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

[28] Shubham Rathi. Generating counterfactual and contrastive explanations using shap. *arXiv preprint arXiv:1906.09293*, 2019.

[29] Eric Rosen, David Whitney, Michael Fishman, Daniel Ullman, and Stefanie Tellex. Mixed reality as a bidirectional communication interface for human-robot interaction. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[30] Eric Rosen, David Whitney, Elizabeth Phillips, Gary Chien, James Tompkin, George Konidaris, and Stefanie Tellex. Communicating robot arm motion intent through mixed reality head-mounted displays. In *Robotics Research*, pages 301–316. Springer, 2020.

[31] Lindsay Sanneman and Julie A Shah. A situation awareness-based framework for design and evaluation of explainable ai. In *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, pages 94–110. Springer, 2020.

[32] Sarah Strohkorb Sebo, Priyanka Krishnamurthi, and Brian Scassellati. "i don't believe you": Investigating the effects of robot trust violation and repair. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 57–65. IEEE, 2019.

[33] Simón C Smith and Subramanian Ramamoorthy. Counterfactual explanation and causal inference in service of robustness in robot control. In *2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 1–8. IEEE, 2020.

[34] Aaron Steinfeld, Terrence Fong, David Kaber, Michael Lewis, Jean Scholtz, Alan Schultz, and Michael Goodrich. Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 33–40, 2006.

[35] Daniel Szafir. Mediating human-robot interactions with virtual, augmented, and mixed reality. In *2019 International Conference on Human-Computer Interaction (HCI)*, pages 124–149. Springer, 2019.

[36] Aaquib Tabrez, Jack Kawell, and Bradley Hayes. Asking the right questions: Facilitating semantic constraint specification for robot skill learning and repair. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[37] Aaquib Tabrez, Shivendra Agrawal, and Bradley Hayes. Explanation-based reward coaching to improve human performance via reinforcement learning. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 249–257. IEEE, 2019.

[38] Aaquib Tabrez, Matthew B Luebbers, and Bradley Hayes. A survey of mental modeling techniques in human–robot teaming. *Current Robotics Reports*, pages 1–9, 2020.

[39] Sahil Verma, John Dickerson, and Keegan Hines. Counterfactual explanations for machine learning: A review. *arXiv preprint arXiv:2010.10596*, 2020.

[40] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. Communicating robot motion intent with augmented reality. In *2018 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 316–324, 2018.