Robot Learning via Human Adversarial Games

Jiali Duan*, Qian Wang*, Lerrel Pinto, C.-C. Jay Kuo and Stefanos Nikolaidis

Abstract— Much work in robotics has focused on "human-in-the-loop" learning techniques that improve the efficiency of the learning process. However, these algorithms have made the strong assumption of a *cooperating* human supervisor that assists the robot. In reality, human observers tend to also act in an *adversarial* manner towards deployed robotic systems. We show that this can in fact improve the robustness of the learned models by proposing a physical framework that leverages perturbations applied by a human adversary, guiding the robot towards more robust models. In a manipulation task, we show that grasping success improves significantly when the robot trains with a human adversary as compared to training in a self-supervised manner.

I. Introduction

We focus on the problem of end-to-end learning for planning and control in robotics. For instance, we want a robotic arm to learn robust manipulation grasps that can withstand perturbations using input images from an on-board camera.

Learning such models is challenging, due to the large amount of samples required. For instance, in previous work [1], a robotic arm collected more than 50K examples to learn a grasping model in a self-supervised manner. Researchers at Google [2] developed an arm farm and collected hundreds of thousands of examples for grasping. This shows the power of parallelizing exploration, while it requires a large amount of resources and the system is unable to distinguish between stable and unstable grasps.

To improve sample efficiency, Pinto et al. [3] showed that robust grasps can be learned using a robotic adversary: a second arm that applies disturbances to the first arm. By training jointly both the first arm and the adversary, they show that this can lead to robust grasping solutions.

This configuration, however, typically requires two robotic arms placed in close proximity to each other. What if there is one robotic arm "in the wild" interacting with the environment, as well as with humans?

One approach could be to have the human act as a teammate, and assist the robot in completing the task. An increasing amount of work [4]–[9] has shown the benefits of human feedback in the robot learning process.

At the same time, we should not always expect the human to act as a collaborator. In fact, previous studies in

* Duan and Wang contributed equally to the work.

Duan and Kuo are with the Department of Electrical and Computer Engineering, University of Southern California, Los Angeles 90089, USA. (e-mail: jialidua@usc.edu, cckuo@sipi.usc.edu).

Wang and Nikolaidis are with the Department of Computer Science, University of Southern California, Los Angeles 90089, USA. (e-mail: {wang215, nikolaid}@usc.edu).

Pinto is with the Robotics Institute, Carnegie Mellon University, Pittsburgh 15213, USA. (e-mail: lerrelp@cs.cmu.edu).
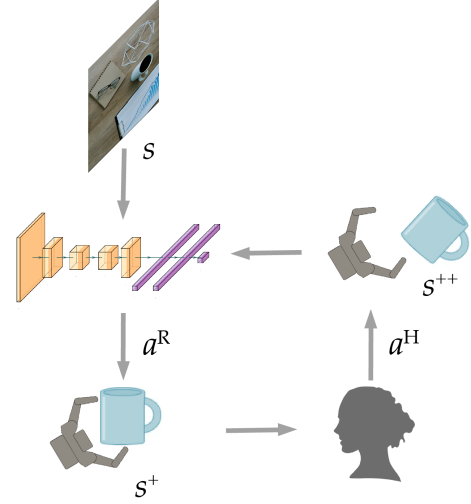
Fig. 1: An overview of our framework for a robot learning robust grasps by interacting with a human adversary.

human-robot interaction [10]–[12] have shown that people, especially children, have acted in an adversarial and even abusive manner when interacting with robots.

This work explores the degree to which a robotic arm could exploit such human adversarial behaviors in its learning process. Specifically, we address the following research question:

> How can we leverage human adversarial actions to improve robustness of the learned policies?

While there has been a rich amount of human-in-the-loop learning, to the best of our knowledge this is the first effort of robot learning with adversarial human users. Our key insight is:

> By using their domain knowledge in applying perturbations, human adversaries can contribute to the efficiency and robustness of robot learning.

We propose an "human-adversarial" framework where a robotic arm collects data for a manipulation task, such as grasping (Fig. 1). Instead of using humans in a collaborative manner, we propose to use them as adversaries. Specifically, we have the robot learner, and the human attempting to make the robot learner fail on its task. For instance, if the learner attempts to grasp an object, the human can apply forces to remove it from the robot. Contrary to a robot adversary in previous work [3], the human has already domain knowledge about the best way to attempt the grasp, by observing the grasp orientation and their prior knowledge of the object's geometry and physics. Additionally, here the robot can only
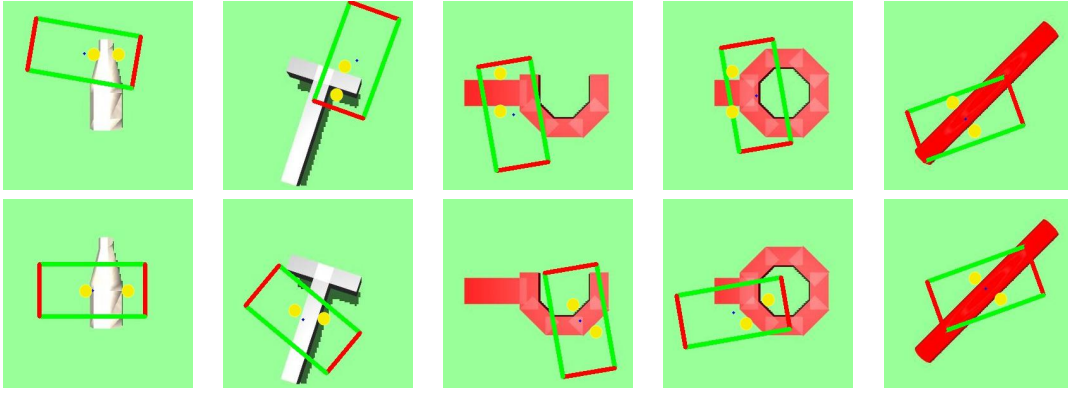
Fig. 2: Selected grasp predictions before (top row) and after (bottom row) training with the human adversary. The red bars show the open gripper position and orientation, while the yellow dots show the grasping points when the gripper has closed.

observe one output, the outcome of the human action, rather than a distribution of adversarial actions.

We implement the framework in a virtual environment, where we allow the human to apply simulated forces on an object grasped by a robotic arm. In a user study we show that, compared to the robot learning in a self-supervised manner, the human user can provide supervision that rejects unstable robot grasps, leading to significantly more robust grasping solutions (Fig. 2).

While there are certain limitations on the human adversarial inputs because of the interface, this is an exciting first step towards leveraging human adversarial actions in robot learning.

II. RELATED WORK

Self-supervised Deep Learning in Manipulation. In robotic manipulation, deep learning has been combined with self-supervision techniques to achieve end-to-end training [2], [13], [14], for instance with curriculum learning [15]. Other approaches include learning dynamics models through interaction with objects [16]. Most relevant to ours is the work by Pinto et al., where a "protagonist" robot learns grasping solutions by interacting with a robotic adversary. In this work, we follow a human-in-the-loop approach, where we have a robotic arm learn robust grasps by interacting with a human adversary.

Reinforcement Learning with Human Feedback. Previous work [4], [7], [17]–[21] has also focused on using human feedback to augment the learning of autonomous agents. Specifically, rather than optimizing a reward function, learning agents respond to positive and negative feedback signals provided by a human supervisor. These works have explored different ways to incorporate feedback into the learning process, either as part of the reward function of the agent, such as in the TAMER framework [20], or directly in the advantage function of the algorithm, as suggested by the COACH algorithm [4]. This allows the human to train the agent towards specific behaviors, without detailed knowledge of the agent's decision making mechanism. Our work is related in that the human affects the agent's reward function. However, the human does not do this explicitly, but indirectly

through its own actions. More importantly, the human acts in an adversarial manner, rather than as a collaborator or a supervisor.

Adversarial Methods. Generative adversarial methods [22], [23] have been used to train two models, a generative model that captures the data distribution, and a discriminative model that estimates the probability that a sample came from the training data. Researchers have also analyzed a network to generate adversarial examples, with the goal of increasing the robustness of classifiers [24]. In our case, we let a human agent generate the adversarial examples that enable adaptation of a discriminative model.

Grasping. We focus on generating *robust* grasps, that can withstand disturbances. There is a large amount of previous work on grasping [25], [26], that range from physics-based modeling [27]–[29] to data-driven techniques [1], [2]. The latter have focused on large-scale data collection. Pinto et al. [3] have shown that perturbing grasps by shaking or snatching by a robot adversary can facilitate learning. We are interested in whether this can hold when the adversary is a human user, applying forces at the grasped object.

III. PROBLEM STATEMENT

We formulate the problem as a *two-player game with incomplete information* [30], played by a human (H) and a robot (R). We define $s \in S$ to be the *state* of the world. A robot and a human are taking turns in actions. A robot action results in a stochastic transition to new state $s^+ \in S^+$, based on some unknown transition function $\mathcal{T} : S \times A^{\mathrm{R}} \rightarrow \Pi(S^+)$. The human then acts based on a stochastic policy, also unknown to the robot, so that $\pi^{\mathrm{H}} : (s^+, a^{\mathrm{H}})$. After the human and the robot's actions, the robot observes the final state s^{++} and receives a reward signal $r : (s, a^{\mathrm{R}}, s^+, a^{\mathrm{H}}, s^{++}) \mapsto r$.

In an adversarial setting, the robot attempts to maximize r, while the human wishes to minimize it. Specifically, we formulate r as a linear combination of two terms: the reward that the robot would receive in the absence of an adversary, and the penalty induced by the human action:

$$r = R^{\mathrm{R}}(s, a^{\mathrm{R}}, s^+) - \alpha R^{\mathrm{H}}(s^+, a^{\mathrm{H}}, s^{++}) \qquad (1)$$

The goal of the system is to develop a policy $\pi^{\text{R}} : s \mapsto a_t^{\text{R}}$ that maximizes this reward.

$$\pi_*^{\text{R}} = \underset{\pi^{\text{R}}}{\arg\max} \; \mathbb{E}\left[r(s, a^{\text{R}}, a^{\text{H}})|\pi^{\text{H}}\right] \qquad (2)$$

Through this maximization, the robot implicitly attempts to minimize the reward of the human adversary. In Eq. (1), α controls the proportion of learning from the human's adversarial actions.

IV. APPROACH

Algorithm. We assume that the robot's policy π^{R} is parameterized by a set of parameters W, represented by a convolutional neural network. The robot uses its sensors to receive a state representation s, and samples an action a^{R}. It then observes a new state s^+, and waits for the human adversary to act. Finally, it observes the final state s^{++}, and computes the reward r based on Eq. (1). A new world state is then sampled randomly, as the robot attempts to grasp a potentially different object (Algorithm 1).

Initialization. We initialize the parameters W by optimizing only for $R^{\text{R}}(s, a^{\text{R}}, s+)$, that is for the reward in the absence of the adversary. This allows the robot to choose actions that have a high probability of grasp success, which in turn enables the human to act in response. After training in a self-supervised manner, the network can be refined through interactions with the human.

Algorithm 1 Learning with a Human Adversary

1: Initialize parameters W of robot's policy π^{R}
2: **for** batch $= 1, B$ **do**
3:     **for** episode $= 1, M$ **do**
4:         observe s
5:         sample action $a^{\text{R}} \sim \pi_*^{\text{R}}(s)$
6:         execute action a^{R} and observe s^+
7:         **if** s^+ is not terminal **then**
8:             observe human action a^{H} and state s^{++}
9:         observe r given by Eq. (1)
10:         record s, a^{R}, r
11:     update W based on recorded sequence
12: return W

V. LEARNING ROBUST GRASPS

We instantiate the problem in a grasping framework. The robot attempts to grasp an object. The human observes the robot's grasp. If the grasp is successful, the human can apply a force as a disturbance in the robot's hand, in six different directions. In this work, we use a *simulation* environment to simulate the grasps and interactions with the human. We use this environment as a testbed for testing different grasping strategies.
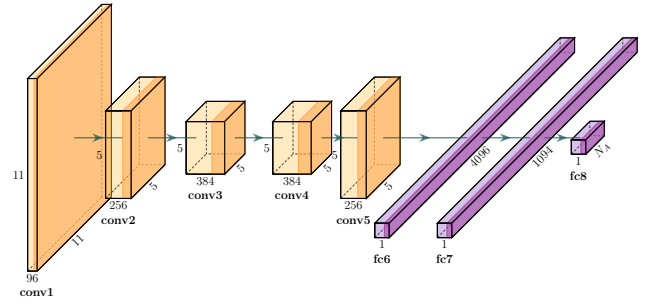

Fig. 3: ConvNet architecture for grasping.

A. Grasping Prediction

Following previous work [1], we formulate grasping prediction as a classification problem. Given a 2D input image I, taken by a camera with a top-down view, we sample N_g image patches. We then discretize the space of grasp angles to N_a different angles. We use the patches as input to a convolutional neural network, which predicts the probability of success for every grasping angle with the grasp location being the center of the patch. The output of the ConvNet is a N_a-dimensional vector giving the likelihood of each angle. This results in a $N_g \times N_a$ grasp probability matrix. The policy then chooses the best patch and angle to execute the grasp. The robot's policy thus uses as input the image I, and as output the grasp location (x_g, y_g), which is the center of the sampled patch, and the grasping angle θ_g: $\pi^{\text{R}} : I \mapsto (x_g, y_g, \theta_g)$.

B. Adversarial Disturbance

After the robot grasps an object successfully, the human can attempt to pull the object away from the robot's end-effector, by applying a force of fixed magnitude. The action space is discrete with 6 different actions, one for each direction: up/down, left/right, inwards/outwards. As a result of the applied force, the object either remains on the robot's hand, or it is dropped to the ground.

C. Network Architecture

We use the same ConvNet architecture with previous work [1], modeled on AlexNet [31] and shown in Fig. 3. The output of the network is scaled to $(0, 1)$ using a sigmoidal response function.

D. Network Training

We initialized the network with a pretrained model initialized by Pinto et al. [1]. The model was pre-trained with completely different objects and patches. To train the model, we treat the reward r that the robot receives as a training target for the network. Specifically, we set $R^{\text{R}}(s, a^{\text{R}}, s^+) = 1$ if the robot succeeds and 0 if the robot fails. Similarly, $R^{\text{H}}(s^+, a^{\text{H}}, s^{++}) = 1$ if the human succeeds, and 0 if the human fails. Therefore, based on Eq. (1), the signal received by the robot is:

$$r = \begin{cases} 0 & \text{if robot fails to grasp} \\ 1 & \text{if robot succeeds and human fails} \\ 1 - \alpha & \text{if human succeeds} \end{cases} \quad (3)$$

We note that the training target is different than that of previous work [3]. There, the robot has access to the adversary's predictions by incorporating into the robot's loss function the probability that the adversarial network believes it can succeed. Here, however, the robot can only observe the outcome of the adversary's action.

We then define as loss function for the ConvNet, the binary cross entropy loss between the network's prediction and the reward received. We train the network using RMSProp [32].

E. Simulation Environment

For the training, we used the Mujoco [33] simulation environment. We customized the environment to allow a human user interacting with the physics engine.[1]

VI. FROM THEORY TO USERS

We conducted a user study, with participants interacting with the robot in the virtual environment. The purpose of our study is to test whether the robustness of the robot's grasps can improve when interacting with a human adversary. We are also interested to explore how the object geometry affects the adversarial strategies of the users, as well as how users perceive robot's performance.

Study Protocol. Participants interacted with a simulated Baxter robot in the customized Mujoco simulation environment (Fig. 4). The experimenter told participants that the goal of the study is to maximize robot's failure in grasping the object. They did not tell participants that the robot was learning from their actions. Participants applied forces to the object using the keyboard. All participants first did a short training phase by attempting to snatch an object from the robot's grasp 10 times, in order to get accustomed to the interface. The robot did not learn during that phase. Then, participants interacted with the robot executing Algorithm 1.

In order to keep the interactions with users short, we simplified the task, so that each user trained with the robot on one object only, presented to the robot at the same orientation. We fixed the magnitude of the forces applied to each object, so that the adversary would succeed if the grasp was unstable but fail to snatch the object otherwise. We selected a batch size $B = 5$ and a number of episodes per batch $M = 9$. The interaction with the robot lasted on average 10 minutes [2].

Manipulated variables. We manipulated (1) the robot's learning framework and (2) the object that users interacted with. We had three conditions for the first independent variable: the robot interacting with a human adversary, the robot interacting with a simulated adversary that learns how

[1]The code is publicly available at: https://github.com/icaros-usc/Interactive-mujoco_py

[2]The anonymized log files of the human adversarial actions are publicly available at: https://github.com/icaros-usc/human_adversarial_grasping_data

1. The robot learned throughout the study.
2. The performance of the robot improved throughout the study.

to succeed in snatching the object and the robot learning in a self-supervised manner, without an adversary. Following previous work [3], the simulated adversary is trained with an identical network with training target equal to 1 if the snatching succeeds and 0 if the snatching fails.

We had five different objects (Fig. 2). We selected objects of varying grasping difficulty and geometry to explore the different strategies employed by the human adversary.

Dependent measures. For testing we executed the learned policy on the object for 50 episodes, applying a random disturbance after each grasp and recording the success or failure of the grasp before and after the random disturbance was applied. To avoid overfitting, we selected for testing the earliest learned model that met a selection criterion (early-stop) [34]. The testing was done using a script after the conduction of the study, without the participants being present. We additionally asked participants to report their agreement on a seven-point Likert scale to two statements regarding the robot's learning process (Table I) and justify their answer.

Hypotheses

H1. *We hypothesize that the robot trained with the human adversary will perform better than the robot trained in a self-supervised manner.* We base this hypothesis on previous work [3] that has shown that training with a simulated adversary improved robot's performance, compared to training in a self-supervised manner.

H2. *We hypothesize that the robot trained with the human adversary will perform better than the robot trained with a simulated adversary.* A human adversary has domain knowledge: they observe the object geometry and have intuition about the physics properties. Therefore, we expect the human to act as a *model-based learning agent* and use their model to do targeted adversarial actions. On the other hand, the simulated adversary has no such knowledge and they need to learn the outcome of different actions through interaction.

Subject allocation. We recruited 25 users, 21 Male and 4 female participants. We followed a between-subjects design, where we had 5 users per object, in order to avoid confounding effects of humans learning to apply perturbations, getting tired or bored by the study.

VII. RESULTS

A. Analysis

Objective metrics. Table II shows the success rates for different objects. Different users interacted with each object; for instance User 1 for Bottle is a different participant than User 1 for T-shape. We have two dependent variables, the success rate of robot grasping an object in the testing phase in the absence of any perturbations, and the success rate with random perturbations being applied. A two-way multivariate ANOVA [35] with object and framework as independent

TABLE II: Grasping success rate (per cent) before (left column) and after (right column) application of random disturbance. Different users interacted with different objects (between-subjects design).

User #	Bottle		T-shape		Half-nut		Round-nut		Stick	
1	64	40	56	42	40	36	58	40	90	62
2	64	40	52	28	40	36	82	48	94	64
3	66	40	56	42	40	36	82	54	92	64
4	74	40	78	60	40	36	52	40	90	62
5	68	40	78	62	40	36	84	48	100	84
Simulated-adversary	60	38	76	54	42	38	54	50	64	54
Self-trained	14	4	52	34	40	36	80	40	50	18

Fig. 4: Participants interacted with a simulated Baxter robot in the customized Mujoco simulation environment.

variables showed a statistically significant interaction effect for both dependent measures: $(F(16, 38) = 3.07, p = 0.002, \text{Wilks'} \Lambda = 0.19)$. In line with **H1**, a Post-hoc Tukey tests with Bonferroni correction showed that success rates were significantly larger for the human adversary condition than the self trained condition, both with $(p < 0.001)$ and without random disturbances $(p = 0.001)$.

We note that the post-hoc analysis should be viewed with caution, because of the significant interaction effect. To interpret these results, we plot the mean success rates for all conditions (Fig. 5). For clarity, we also contrast both success rates for each object separately in Fig. 6. Indeed, we see the the success rate averaged over all human adversaries was higher for three out of five objects. The difference was largest for the bottle and the stick. The reason is that it was easy for the self-trained policy to pick up these objects without a robust grasp, which resulted in slow learning. On the other hand, the network trained with the human adversary rejected these unstable grasps, and learned quickly robust grasps for these objects. In contrast, round nut and half-nut objects could be grasped robustly at the curved areas of the object. The self-trained network thus got "lucky" finding these grasps, and the difference was negligible. In summary, these results lead to the following insight:

> Training with a human adversary is particularly beneficial for objects that have few robust grasp candidates that the network needs to search for.

There were no significant differences between the rates in the human adversary and simulated adversary condition.

Indeed, we see that the mean success rates were quite close for the two conditions. We expected the human adversary to perform better, since we hypothesized that the human adversary has a *model* of the environment, which the simulated adversary does not have. Therefore, we expected the human adversarial actions to be more targeted. To explain this result, which does not support **H2**, we look at human behaviors below.

Behaviors. Fig. 7 shows the disturbances applied over time for different users. Observing the participants behaviors, we see that some participants *used their model of the environment to apply disturbances effectively*. Specifically, the user in Fig. 7(b) applied a force outwards in the T-shape, succeeding in 'snatching' the object even at the first try, which is indicated by the red dots. Gradually, the robot learned a more robust grasping policy, which resulted in the user failing to snatch the object (green dots). Similarly, the user in Fig. 7(a) and Fig. 7(c) used targeted perturbations which resulted in failed grasps from the very start of the task.

In some cases, such as in Fig. 7(e), the user adapted their strategy as well: when the robot learned to withstand an adversarial action outwards, the user acted by applying a force to the right, until the robot learned that as well.

Fig. 8 compares the user of Fig. 7(e) with the simulated adversary for the same object (stick). We observe that the simulated adversary explores different perturbations that are unsuccessful in snatching the object. This translates to worse performance for that object in the testing phase.

However, not all grasps required an informed adversary for the grasp to fail. For instance, for the grasped bottle in Fig. 9(a), there were many different directions where an applied force could succeed in removing the object. Therefore, having a model of the environment did not offer a significant benefit, since almost any disturbance would succeed in dropping the object. On the contrary, several grasps of the stick object failed only with targeted disturbances in the direction parallel to the object's major axis (Fig. 9(b)), which explains the difference in performance between human and simulated adversaries for that object.

Additionally, we found that some participants did not act as rational, model-based agents, which is the second factor that we believe affected the results. For instance, looking at one of the participants' interactions with the stick object (Fig. 10), we see the variance of the actions increasing over time. We found this variance surprising, given the geometry of the object and the fact that all subsequent perturbations
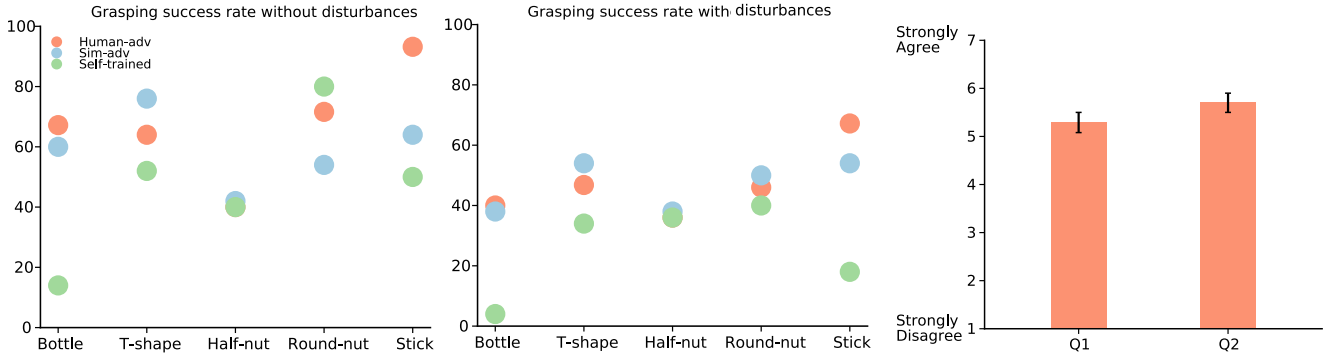
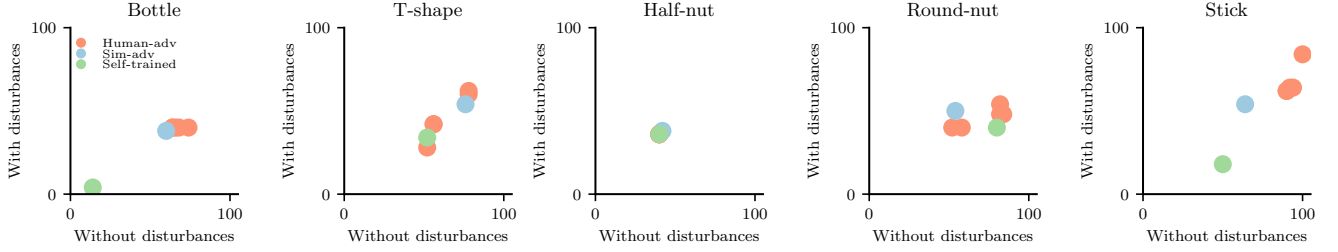Fig. 5: Success rates from Table II for all five participants and subjective metrics.

Fig. 6: Success rates from Table II for each object with (y-axis) and without (x-axis) random disturbances for all five participants.
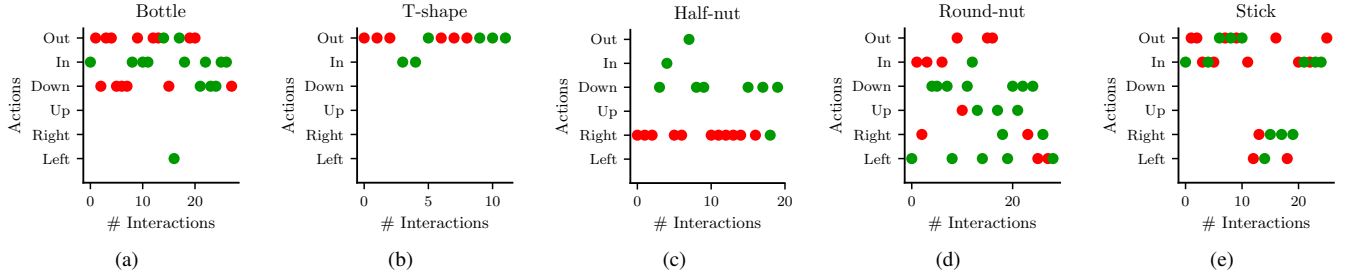

Fig. 7: Actions applied by selected human adversaries over time. We plot in green adversarial actions that the robot succeeds in resisting and in red actions that result in the human 'snatching' the object.
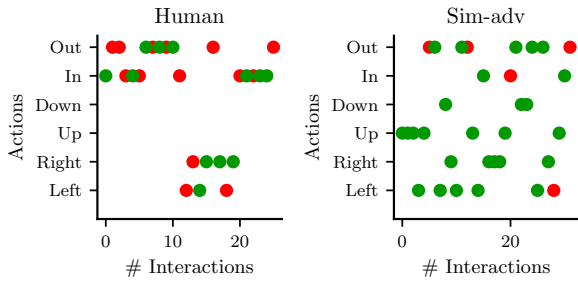

Fig. 8: Difference between training with user and simulated adversary for the stick object. The simulated adversary explores by applying forces in directions that fail to snatch the object. The red dots indicate human success in snatching the object, while the green dots indicate robot success in withstanding the human perturbation.
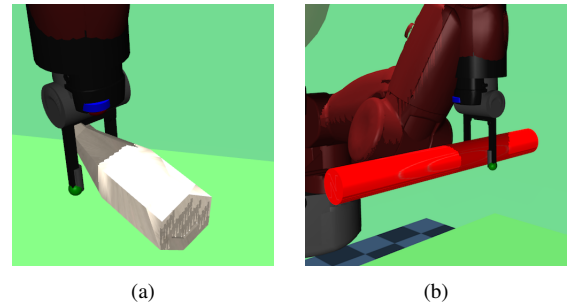

Fig. 9: A force in almost any direction would make the grasp (a) fail, while only a force parallel to the axis of the stick would snatch the object in grasp (b).

were unsuccessful. Looking at the open-ended responses, the participant stated that "it seems some perturbations were challenging; so after some time I didn't apply that perturbation again." This indicates that at least one participant did not follow our instructions to act in an adversarial manner,

and wanted to assist the robot instead.

Subjective metrics. We conclude our analysis with reporting the users' subjective responses (Fig. 5). A Cronbach's $\alpha = 0.86$ showed good internal consistency [36]. Participants generally agreed that the robot learned throughout the study, and that its performance improved. In their open-ended
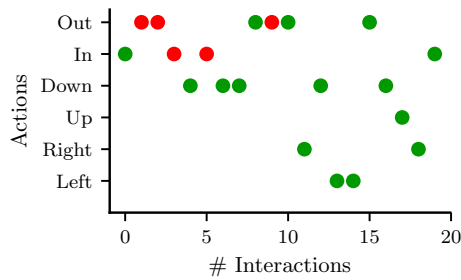
Fig. 10: The user started assisting the robot in the later part of the interaction, instead of acting as an adversary. The red dots indicate human success in snatching the object, while the green dots indicate robot success in withstanding the human perturbation.

responses, participants stated that "The robot learned the grasping technique to win over me by learning from the forces that I provided and became more robust," and that "The robot took almost 8 to 10 runs before it would start responding well. By the end of my experiment, it would grasp almost all the time." At the same time, one participant stated that the "rate of improvement seemed pretty slow," and another that it "kept making mistakes even towards the end."

B. Multiple objects.

We wish to test whether our framework can leverage human adversarial actions to learn grasping multiple objects at the same training session. Therefore, we modified the experiment setup, so that in each episode one of the five objects appeared randomly. To increase task difficulty, we additionally randomized the object's position and orientation in every episode. The robot then trained with one of the authors of the paper for 200 episodes. We then tested the trained model for another 200 episodes with randomly selected objects of random positions and orientations, as well as randomly applied disturbances. The trained model achieved a 52% grasping success rate without disturbances, and 34% success rate with disturbances. The rates were higher than those of a simulated adversary trained in the same environment for the same number of episodes, which had 28% grasping success rate without disturbances and 22% with disturbances. We find this result promising, since it indicates that targeted perturbations from a human expert can improve the efficiency and robustness of robot grasping.

VIII. CONCLUSION

Limitations. Our work is limited in many ways. Our experiment was conducted in a virtual environment, and the users' adversarial actions were constrained by the interface. Our environment provides a testbed for different human-robot interaction algorithms in manipulation tasks, but we are also interested in exploring what types of adversarial actions users apply in real-world settings. We also focused on interactions with only one human adversary; a robot "in the wild" is likely to interact with multiple users. Previous work [3] has shown that training a model with different robotic adversaries

further improves performance, and it is worth exploring whether the same holds for human adversaries.

Implications. Humans are not always going to act cooperatively with their robotic counterparts. This work shows that from a learning perspective, this is not necessarily a bad thing. We believe that we have only scratched the surface of the potential applications of learning via adversarial human games: Humans can understand stability and robustness better than learned adversaries, and we are excited to explore human-in-the-loop adversarial learning in other tasks as well, such as obstacle avoidance for manipulators and mobile robots.

REFERENCES

[1] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3406–3413.

[2] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[3] L. Pinto, J. Davidson, and A. Gupta, "Supervision via competition: Robot adversaries for learning tasks," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1601–1608.

[4] J. MacGlashan, M. K. Ho, R. Loftin, B. Peng, D. Roberts, M. E. Taylor, and M. L. Littman, "Interactive learning from policy-dependent human feedback," *arXiv preprint arXiv:1701.06049*, 2017.

[5] G. Warnell, N. Waytowich, V. Lawhern, and P. Stone, "Deep tamer: Interactive agent shaping in high-dimensional state spaces," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[6] 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*. ACM, 2009, pp. 9–16.

[7] ——, "Reinforcement learning from simultaneous human and mdp reward," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2012, pp. 475–482.

[8] Z. Lin, B. Harrison, A. Keech, and M. O. Riedl, "Explore, exploit or listen: Combining human feedback and policy model to speed up deep reinforcement learning in 3d worlds," *arXiv preprint arXiv:1709.03969*, 2017.

[9] S. Reddy, S. Levine, and A. Dragan, "Shared autonomy via deep reinforcement learning," *arXiv preprint arXiv:1802.01744*, 2018.

[10] C. Bartneck and J. Hu, "Exploring the abuse of robots," *Interaction Studies*, vol. 9, no. 3, pp. 415–433, 2008.

[11] D. Brscić, H. Kidokoro, Y. Suehiro, and T. Kanda, "Escaping from children's abuse of social robots," in *Proceedings of the tenth annual acm/ieee international conference on human-robot interaction*. ACM, 2015, pp. 59–66.

[12] T. Nomura, T. Kanda, H. Kidokoro, Y. Suehiro, and S. Yamada, "Why do children abuse robots?" *Interaction Studies*, vol. 17, no. 3, pp. 347–369, 2016.

[13] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[14] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.

[15] L. Pinto and A. Gupta, "Learning to push by grasping: Using multiple tasks for effective learning," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2161–2168.

[16] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine, "Learning to poke by poking: Experiential learning of intuitive physics," in *Advances in Neural Information Processing Systems*, 2016, pp. 5074–5082.

[17] W. B. Knox and P. Stone, "Combining manual feedback with subsequent mdp reward signals for reinforcement learning," in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1.* International Foundation for Autonomous Agents and Multiagent Systems, 2010, pp. 5–12.

[18] R. Loftin, B. Peng, J. MacGlashan, M. L. Littman, M. E. Taylor, J. Huang, and D. L. Roberts, "Learning behaviors via human-delivered discrete feedback: modeling implicit feedback strategies to speed up learning," *Autonomous agents and multi-agent systems*, vol. 30, no. 1, pp. 30–59, 2016.

[19] S. Griffith, K. Subramanian, J. Scholz, C. L. Isbell, and A. L. Thomaz, "Policy shaping: Integrating human feedback with reinforcement learning," in *Advances in neural information processing systems*, 2013, pp. 2625–2633.

[20] W. B. Knox and P. Stone, "Tamer: Training an agent manually via evaluative reinforcement," in *2008 7th IEEE International Conference on Development and Learning.* IEEE, 2008, pp. 292–297.

[21] D. Arumugam, J. K. Lee, S. Saskin, and M. L. Littman, "Deep reinforcement learning from policy-dependent human feedback," *arXiv preprint arXiv:1902.04257*, 2019.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[23] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[24] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[25] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[26] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1. IEEE, 2000, pp. 348–353.

[27] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2016, pp. 1957–1964.

[28] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *2007 7th IEEE-RAS International Conference on Humanoid Robots.* IEEE, 2007, pp. 42–48.

[29] D. Berenson and S. S. Srinivasa, "Grasp synthesis in cluttered environments for dexterous hands," in *Humanoids 2008-8th IEEE-RAS International Conference on Humanoid Robots.* IEEE, 2008, pp. 189–196.

[30] R. Lavi, "Algorithmic game theory," *Computationally-efficient approximate mechanisms*, pp. 301–330, 2007.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[32] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[33] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.* IEEE, 2012, pp. 5026–5033.

[34] R. Caruana, S. Lawrence, and C. L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Advances in neural information processing systems*, 2001, pp. 402–408.

[35] M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li *et al.*, *Applied linear statistical models.* McGraw-Hill Irwin Boston, 2005, vol. 103.

[36] J. M. Bland and D. G. Altman, "Statistics notes: Cronbach's alpha," *Bmj*, vol. 314, no. 7080, p. 572, 1997.