2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Learning Reward Functions for Robotic Manipulation by Observing Humans

Minttu Alakuijala[1,2], Gabriel Dulac-Arnold[3], Julien Mairal[2], Jean Ponce[1,4] and Cordelia Schmid[3]

***Abstract*— Observing a human demonstrator manipulate objects provides a rich, scalable and inexpensive source of data for learning robotic policies. However, transferring skills from human videos to a robotic manipulator poses several challenges, not least a difference in action and observation spaces. In this work, we use unlabeled videos of humans solving a wide range of manipulation tasks to learn a task-agnostic reward function for robotic manipulation policies. Thanks to the diversity of this training data, the learned reward function sufficiently generalizes to image observations from a previously unseen robot embodiment and environment to provide a meaningful prior for directed exploration in reinforcement learning. We propose two methods for scoring states relative to a goal image: through direct temporal regression, and through distances in an embedding space obtained with time-contrastive learning. By conditioning the function on a goal image, we are able to reuse one model across a variety of tasks. Unlike prior work on leveraging human videos to teach robots, our method, Human Offline Learned Distances (HOLD) requires neither a priori data from the robot environment, nor a set of task-specific human demonstrations, nor a predefined notion of correspondence across morphologies, yet it is able to accelerate training of several manipulation tasks on a simulated robot arm compared to using only a sparse reward obtained from task completion.**

I. Introduction

Deep learning has greatly advanced the state of the art in applications ranging from computer vision [1], [2] to natural language processing [3], [4] to speech recognition [5], but its significance in robotics has been blunted by limited access to large-scale data. Although previous efforts have covered a specific embodiment and task [6], [7], collecting a massive dataset for each robot and environment of interest is simply not feasible due to the cost of maintenance, human oversight, hardware wear and tear and the bottleneck of real-time execution. For these reasons, creative reuse of data is of central importance for unlocking the benefits of large-scale data-driven learning in robotics.

One potential source of external data is videos of humans performing arbitrary tasks, widely available on the internet and inexpensive to produce. We focus on manipulation tasks in this work, with the aim of learning from crowd-sourced videos of human arms and hands. However, replicating the demonstrated actions and object interactions with a robot is a challenging open problem. On the perception side, there is a significant visual domain gap between observations of a person and of a robot. Human and robot arms usually have very different morphologies and dynamics, particularly in the end-effector, creating a physical domain gap and making a 1:1 mapping between poses ill-defined in general. Moreover, the actions taken by humans are not observed unless explicitly recorded with specialized equipment, and hence conventional imitation learning [8], [9] or offline reinforcement learning [10], [11] methods are not applicable.

To overcome these challenges, we investigate the use of videos of people solving manipulation tasks to learn a notion of distance between images from the observation space of a task. We leverage this learned distance as a reward signal on tasks with similar structure but very different visual appearance on a set of robotic manipulation domains that the model has never observed. By training on diverse human demonstrations, we employ a strategy analogous to domain randomization [12] used in sim-to-real transfer, which applies variations to visual and physical simulation parameters at training time so that a real-world robotic task with unknown physical properties is more likely to fall in the training distribution. Similarly, when trained with different demonstrators, backgrounds, viewpoints, lightings, objects and tasks, our distance model learns to generalize to a variety of manipulator appearances. Furthermore, several aspects of the task as solved by a human are preserved in the robot workspace. For example, object displacements must respect the laws of physics regardless of the actor.

The learned distance function captures roughly how long it takes for an expert to transition from one state to another, and is therefore closely related to a dense reward function representing task progress that can be optimized with reinforcement learning (RL). Learning dense rewards is especially useful in hard exploration problems where it is straightforward to define a sparse task-completion reward, but laborious and error-prone to specify a well-shaped dense reward.

Instead of model-free RL, reward functions estimating task progress can also be optimized with model predictive control [13], [14], in which case both a forward model of the environment dynamics and a state-action value function need to be learned, typically from undirected exploration data in the target environment. However, these methods require extensive a priori data collection with sufficient coverage on a target robot environment and its action space, and learning accurate video prediction models remains a challenging open problem in itself. We instead propose to learn a state-value

[1]Département d'informatique de l'Ecole normale supérieure (ENS-PSL, CNRS, Inria) {minttu.alakuijala,jean.ponce}@inria.fr
[2]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France julien.mairal@inria.fr
[3]Google Research {dulacarnold,cordelias}@google.com
[4]Courant Institute of Mathematical Sciences and Center for Data Science, New York University

function from observation-only data which allows for the reuse of data from different embodiments, and train a policy for the target embodiment with online RL. We empirically show better sample efficiency per task in online training than was required to learn a model in prior work [13].

Our contributions are as follows: i) We propose HOLD[1], a global goal-conditioned distance model which removes the need for demonstration task labels and exact alignment between robot tasks and demonstrated tasks required by prior work [13], [15]–[20]. ii) We show generalization of reward functions trained on unconstrained human videos to robot arms of various morphologies and environments, and accelerate training of model-free RL on 5 simulated manipulation tasks by up to 18x by providing shaped rewards in sparse-reward tasks, or even entirely replacing the reward in some tasks. iii) We show that time-contrastive embeddings [21] can successfully represent distances for multiple tasks at once despite a high degree of multi-modality in mixed-task training data. iv) We show HOLD to outperform existing cross-domain imitation [21] and representation learning [22] approaches able to handle mixed-task videos.

II. Related work

a) Intermediate representations: Several prior works have addressed learning robotic policies from human videos via intermediate representations such as pose estimation or keypoint tracking [23]–[25]. In this work, our aim is to advance the capabilities of learning from raw video data, without depending on hand-crafted intermediate representations of human hands or an object database.

b) Imitation learning: Our work is also related to imitation learning from observation, although this line of work has mostly addressed the case of demonstrations from the same observation space [9], [26]–[28]. We instead tackle the more difficult problem of inverse RL from observation under significant observational and dynamical domain shift.

c) Offline RL: Similarly to HOLD, Offline RL [11], [29]–[31] also aims to learn a value function from a dataset of existing trajectories. However, our setting is significantly different from the offline RL problem as we do not have access to either the actions or the rewards of the demonstrator in our dataset, nor do we have a forward model of which states are reachable from a given state, making temporal difference based methods not applicable.

d) Mapping methods: Many methods for learning from videos seek to learn a direct mapping between demonstration videos and robot states and/or actions, such as an inverse model labeling each human transition with an action from the robot action space [15], or an image-to-image translation of a human demonstration to a corresponding robot demonstration [32], [33]. By contrast, our method does not assume a precise 1:1 mapping between the observation and action spaces of the human and the robot and can therefore leverage arbitrarily large amounts of human demonstration videos without any manual supervision cost.

[1]Code and videos are available on sites.google.com/view/hold-rewards.

e) Consistency methods: A line of prior work has proposed to learn domain-invariant features capturing task progress regardless of whether the actor is a human or a robot arm [15], [16], [21] with reward usually defined as distance to a human demonstration [21] or to a goal state [16] in the feature space. One issue with using geometrical distances is that transition times between states are not symmetrical if the environment includes unidirectional transitions, such as dropping an object or knocking something down. To account for this, we additionally propose a regression-based model which predicts distances as a function of two ordered states. Sequence-based objectives such as temporal cycle consistency [16] are well suited for single-task learning where all trajectories can be aligned along a global task progression, but it is unclear whether these methods would work on data from several tasks. Most existing approaches to learning robotic manipulation from human videos also require either exact overlap between tasks demonstrated by humans and the robot tasks [13], [15]–[17] or robot demonstrations for many of the same tasks [13], [18]–[20]. As our model is not specialized for any single task and learns from human data only, no robot demonstrations are needed and the target robot task does not need to be strictly included in the training data as long as a goal image is available to specify the new task.

f) Time-contrastive embeddings: Sermanet *et al.* [21] propose to use distances in an embedding space learned with a time-contrastive objective, but only consider reward learning for a single task, whereas we learn a single multi-task reward model. Moreover, while [21] propose to directly imitate a human demonstration at 1:1 speed, we instead define the task with a goal image from the robot's observation space. As we show experimentally, [21] needs a nearly identical alignment in the initial states, execution speed and cropping between the video and the robot observations, which is a significant limitation. By contrast, our inverse RL approach requires less supervision and allows the robot to potentially outperform the demonstrator, either by executing the task faster or by finding a more optimal trajectory. Concurrently to our work, Ma *et al.* [34] use implicit time-contrastive learning to train a task-agnostic visual reward function related to our time-contrastive model. However, their approach also does not consider the potential asymmetry of dynamics that our regression model can represent.

g) Functional distance: Our work is also related to estimating functional (also called dynamical) distance between states from online [35] or offline robot data [14]. However, both works use only robot data from the same environment, without transfer of the action or observation spaces. Our approach is instead based on estimating the state-value function of the demonstrated behavior drawn from an unknown action space.

III. Human Offline Learned Distances

A. Functional distances from observation-only data

We propose to learn about distances in state space by observing humans and using this prior knowledge of environment dynamics to accelerate training of robotic manipulation

policies. Specifically, our goal is to estimate *functional distance* $d(s, g)$ [14], [35], between an image s of the current state and a goal image g, where $s, g \in \mathcal{S}_r$, the set of camera images from the robot's observation space. This metric should correlate with $\delta(s, g)$, the number of time steps it takes for an expert policy π^* to reach the goal g from the state s:

$$\delta(s, g) = \mathbb{E}[T | s_T = g, s_0 = s, a_t \sim \pi^*(s_t, g), \\ s_{t+1} \sim p(s_t, a_t)], \tag{1}$$

where p are the transition dynamics of the environment, modeled as a Markov Decision Process (MDP). We assume each image observation fully captures the environment state in order to unambiguously define tasks using goal images. The negated time difference $-\delta(s, g)$ is equal to the value function V^* for an optimal policy π^* and a reward of -1 per time step until the episode terminates (upon successfully reaching the goal or exceeding a time limit). However, this is not the only reward that can be optimized to recover π^* (Ng *et al.* [36] discuss conditions for policy-invariant reward shaping in the general case). Given the original reward $r = -1$ with $V^* = -\delta$, π^* is unchanged for the reward $r'(s_t, a_t, s_{t+1}, g) = -|d(s_{t+1}, g) - d(s_t, g)|$ with $V^* = -d$ if we assume that $d(g, g) = 0$, p is deterministic and pairwise rankings are preserved:

$$\forall s, s', g \in \mathcal{S}_r, \delta(s, g) > \delta(s', g) \implies d(s, g) > d(s', g) \\ \text{and } \delta(s, g) = \delta(s', g) \implies d(s, g) = d(s', g).$$

Although defined in terms of an expert policy π^*, $\delta(s, g)$, and consequently the functions $d(s, g)$ that preserve its rankings, can be estimated from observation-only data, without access to actions a, the expert π^*, or even its action space, by obtaining self-supervised time deltas without manual annotation. While Tian *et al.* [14] learn the Q-function corresponding to a related, sparse goal-reaching reward from offline trajectories from the robot, our choice of a state-value function, agnostic to a specific action space, allows reuse of data gathered with different but related morphologies, such as other robots or humans. Strictly speaking, the ability to share the function δ between human and robot MDPs relies on them being isomorphic [15], requiring a 1:1 mapping between the action and observation spaces that preserves dynamics p. While this may not fully hold in practice, and the distribution of δ in human data may not necessarily match the robot's dynamics in absolute terms due to embodiment differences, the rankings produced by d can be transferred under fewer assumptions. For example, one embodiment may be twice as fast as the other while still preserving all pairwise rankings.

We assume access to a dataset of N video demonstrations of humans executing a variety of manipulation tasks using approximately shortest paths. In practice, the precise length of time may vary significantly across trials and human demonstrators, and depend on the optimality of the demonstration. Although the absolute length of such time intervals may not be consistent across demonstrators, their relative durations provide a useful learning signal; in order to push an object to the right, one must first approach its current position from the left before starting the pushing maneuver, and not the other way around. We present two methods for learning d on this data.

a) Direct regression (HOLD-R): We assume the demonstrations are optimal and pose the functional distance learning problem as a supervised regression task:

$$\theta^* = \text{argmin} \sum_{i=1}^{N} \sum_{t=1}^{T_i} \sum_{\delta=1}^{T_i - t} ||d_\theta(s_t^i, s_{t+\delta}^i) - \delta||_2^2 \tag{2}$$

where s_t^i is the tth frame of the ith video, T_i is the length of the ith video, and d_θ is a function parameterized by θ trained to predict δ from Eq. (1). The third summation corresponds to data augmentation allowing any future time step in the video to be considered the goal rather than only the last.

b) Time-contrastive embeddings (HOLD-C): Since directly predicting time intervals is difficult and sensitive to noise, we may also consider learning an embedding space where distances can be defined. We propose to use a single-view time-contrastive objective as in TCN [21]. Frames within a small temporal window are encouraged to lie close together in embedding space, whereas embeddings for frames outside some temporal neighborhood are pushed apart. Specifically, if s_p is a positive instance for anchor s, and s_n is a negative instance, for all triplets, we want:

$$||f(s) - f(s_p)||_2^2 + m < ||f(s) - f(s_n)||_2^2 \tag{3}$$

where the margin m is a hyperparameter. However, unlike the single-task setup proposed in [21], we train f on multi-task data and show it to accelerate robot learning across tasks. Moreover, our method improves upon TCN in several ways at the policy training stage: i) HOLD enables the robot to outperform the demonstrations by learning relevant shortcuts through interaction, or by simply moving faster, whereas TCN aims to imitate the human. TCN defines the task using a human video, and minimizes distance to each of its states at 1:1 speed – although the distances are minimized with RL, the best possible reward is defined as matching the human performance. ii) HOLD requires less supervision: TCN needs one human trajectory of the full task whereas we use distance to a goal image only and require no task demonstrations. iii) We use a simpler Euclidean distance to define the metric $d(s, g)$ in the space f, whereas [21] apply a weighted mixture of squared Euclidean and a Huber-style loss $d(s_t, g_t) = \alpha||f(s_t) - f(g_t)||_2^2 + \beta\sqrt{\gamma + ||f(s_t) - f(g_t)||_2^2}$, requiring two additional hyperparameters to be tuned in an already computationally expensive RL training setup.

B. Policy learning

We propose to use the learned functional distance to define a dense reward function for an RL policy. Although our reward function is goal-conditioned and shared across tasks, we train one policy per robot task, in order to focus on multi-task reward learning in this work. As we want to minimize distance to the goal frame, we define reward as follows:

$$r(s_t, a_t, s_{t+1}, g) = -\max(0, d(s_{t+1}, g) - d(g, g))/T \tag{4}$$

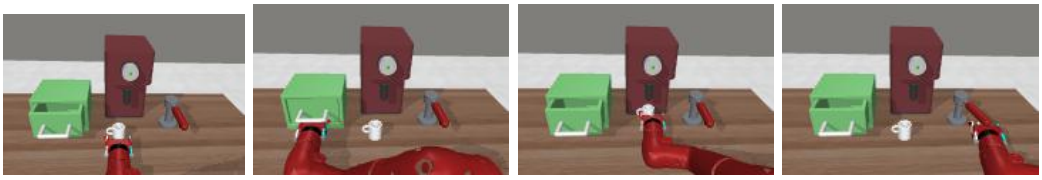
(a) Pushing mouse from left to right

(b) Putting paint brush underneath magazine

(c) Moving book up

Fig. 1: Example human videos from Something-Something v2 used to train the distance models.

(a) Pushing: start (b) Pushing: goal (c) Drawer: start (d) Drawer subtask (e) Drawer: goal



(f) Start state for all tasks (g) Close Drawer (h) Push Cup Forward (i) Turn Faucet Right

Fig. 2: (a-e) The RLV Tasks. (f-i) The DVD tasks: a Sawyer arm in a tabletop environment adapted from Meta-World [37].

where $s_t, s_{t+1}, g \in \mathcal{S}_r$, a_t is an action from the robot's action space, and T is an optional normalizer. We subtract the baseline $d(g, g)$ from the distance estimates to ensure arriving at the goal has reward 0 and no other state has higher reward; $d(g, g)$ may be positive for the regression models due to untrimmed training videos where the demonstrator idles after solving the task. This definition of reward corresponds to minimizing the sum of distances until the end of the episode, as done by [14], [35]. Alternatively, r could be defined based on the difference $d(s_{t+1}, g) - d(s_t, g)$, such that only the reduction is maximized for each time step. However, we found the cumulative form to perform better empirically, possibly due to being less noisy.

IV. Experimental Results

A. Distance learning

a) Dataset: We train HOLD on Something-Something v2 (SSv2) [38], a crowd-sourced dataset of 220,847 video clips of 174 action classes (with examples in Fig. 1). Each action is demonstrated with arbitrary objects, matching templates such as *Moving [something] closer to [something]*. The clips last 4 seconds on average and are mostly filmed using handheld devices, with non-negligible camera motion. Although SSv2 videos are grouped into discrete action classes, we do not make use of these labels[2], making our method applicable on any large-scale goal-oriented manipulation data. As we train a single goal image-conditioned distance function, there also does not need to be exact overlap between the demonstrated tasks and the target tasks on the robot, unlike in prior works [13], [15]–[20].

b) Training details: We consider two sizes of network architecture: a ResNet-50 [1] and a Video Vision Transformer (ViViT) [40] pretrained on SSv2 classification. As the single-view time-contrastive objective only supports embedding single images, for HOLD-C we instead use either a ResNet or a Vision Transformer (ViT) [2] pretrained on ImageNet-21K. We train the ResNet models from scratch, and fine-tune the pretrained models on SSv2 without labels after replacing their classification heads. To adapt the pretrained ViViT model for regression, we also reinitialize its temporal position embeddings and shorten the temporal window to 4, including the 3 most recent frames and one goal frame. We also reduce the temporal filter dimension to 1 as there is no longer a computational benefit to shortening the sequence length. For time-contrastive training, we sample batches of 32 subsequent frames per video and use a positive window of 0.2 seconds and a negative window of 0.4s, as done by [21]. For both objectives, we apply the same data augmentation procedure as [40], but leave out MixUp. Other hyperparameters are included in Table I. We observed better policy training performance for the ResNet model for HOLD-C, and for ViViT for HOLD-R, so we report results using these backbones in Section IV-B.

B. Policy Learning

To demonstrate the utility of our method as a reward function for training RL policies, we evaluate it on the Pushing and Drawer Opening tasks from RLV [15] (Fig. 2a–2e) and on the Close Drawer, Push Cup Forward and

[2]Only HOLD-R with ViViT architecture made use of labels in pretraining, whereas HOLD-R with ResNet backbones as well as all HOLD-C models did not. However, the pretraining could have potentially been done on a different labeled dataset such as Kinetics [39] or skipped altogether, and no labels are needed for the regression task.

Turn Faucet Right tasks from DVD [13] (Fig. 2f–2i). We follow prior work [15], [16] in using Soft Actor-Critic (SAC) [41] as the underlying RL algorithm and evaluate it on 20 episodes for all tasks. All policies use images as input, and we reuse the policy and critic architectures as well as algorithm hyperparameters from [15]. Like [15], we augment our learned reward from Eq. (4) with a sparse task reward: 1 for success, and 0 otherwise, defined by each environment based on distance to the target configuration. Since the predicted distances can be significantly larger than 1 but should not override the sparse reward, we scale the predicted rewards by $1/T$, where T is set such that the scale of initial state distances is $\approx 1/3$.

1) RLV tasks: As shown in Fig. 3, the sum of both reward functions, appropriately balanced, significantly accelerates training compared to using the sparse reward alone. In our experiments, using only sparse reward required 10x more samples for Pushing and >18x more for Drawer to reach the return of HOLD-C. We find that HOLD-C outperforms HOLD-R for Pushing, both with and without sparse reward, and in the sparse reward setting for Drawer Opening.

a) Pushing: Without added sparse reward, a single failure case is prominent: while the policy quickly learns to match the end-effector position in the goal frame, it fails to pay attention to the puck position. As observed by Tian *et al.* [14], it is easy for the distance function to excessively focus on fully actuated components in the scene as these are highly predictive of temporal offset. Although HOLD is able to generalize from human arms to a robot arm, for tasks with variable object positions, it may be better suited as an exploration strategy used together with an otherwise rarely-observed sparse reward than a standalone multi-task reward. Note that although Zakka *et al.* [16] also evaluate on Pushing, their results are not comparable as their method is trained on the easier RLV Pushing dataset [15] collected to match the appearance of the robot task, and report on the simpler State Pusher task where the policy directly observes the 2D puck position and the 3D end-effector position.

b) Drawer: The Drawer Opening task has double the episode length (200 steps) of Pushing, and consists of two distinct motions: approaching and inserting the gripper into the handle, and pulling the drawer open once there. We find that applying the HOLD models on the full task suffers from the local minimum of only matching the arm position in the goal image. However, if we instead define the task in two parts using an intermediate goal image (in Fig. 2d), our rewards significantly improve sample efficiency compared to the sparse task-completion reward provided by the environment alone, as shown in Fig. 3b. Moreover, HOLD-R alone without any environment reward performs on par with sparse reward in this setting. We train a single policy for both subtasks, which is conditioned on the active goal image by concatenating it to the observation s. For all distance functions, we switch to the next subtask when $d < 1$ for at least 3 consecutive time steps.

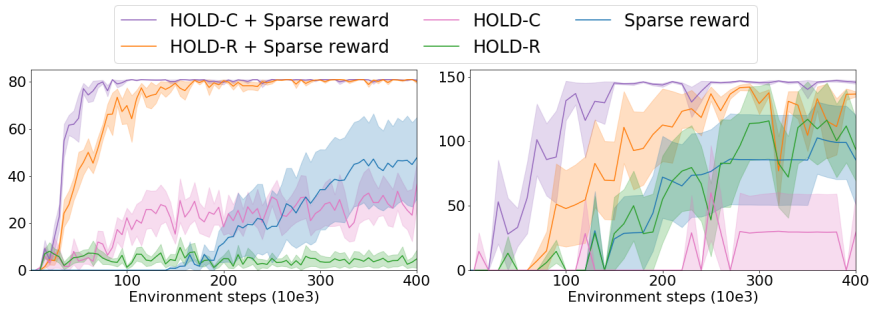
2) DVD tasks: We report success rate for the DVD tasks in Fig. 4. These tasks are significantly easier than the RLV tasks and quickly learned even using only sparse reward. To estimate the upper bound in learning speed achievable by improving the reward alone, we define an oracle reward using knowledge of robot and object positions. Since we observe only a narrow performance gap between the oracle and the sparse reward, the learning speed in these tasks is limited mostly by the RL algorithm. Although it is therefore difficult to show much improvement over the sparse reward, both HOLD models outperform it, particularly for Close Drawer and Turn Faucet Right. For Close Drawer, HOLD also solves the task without sparse reward. Unlike [13], we do not first collect a dataset of 10,000 trajectories, or 600,000 steps, of random exploration on the robot to learn a model of the environment, but instead focus on the model-free setting. We show adaptation to a new robot, set of objects and environment in just 12,000–18,000 steps, or 200 to 300 episodes, when sparse environment reward is available, or 22,000 without sparse reward for Close Drawer.

C. Baseline comparisons

We compare HOLD to rewards defined by two prior methods: TCN [21] and R3M [22]. We note that single-task learning methods RLV [15] and XIRL [16] are not applicable to our setting as they require demonstration task labels. TCN proposes to transfer a policy given a human video demonstration by minimizing distance to the embeddings of each of the visited states g_t in turn. We empirically set the hyperparameters of $d(s_t, g_t)$ to $\alpha = 0.005, \beta = 0.02$, and $\gamma = 0.2$. As performance may vary based on the exact demonstration video used, we evaluate 3 demonstrations per task from the RLV dataset, which is collected to closely match the RLV robot tasks, and report the average performance across demonstrations (trained with 5 seeds each) in Figure 5. Even the closely aligned demonstrations transfer poorly to policy learning, especially for Pushing, due to slight differences in initial state, cropping or execution speed, highlighting the brittleness of the trajectory-following objective of TCN.

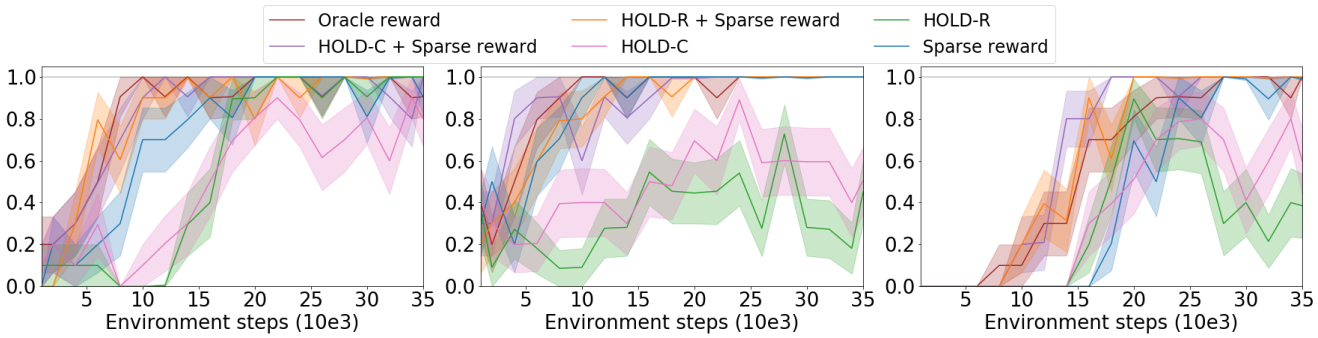
Although R3M is proposed as a general feature representation, we also compare against using Euclidean distance in the representation space for defining dense rewards. We used the ResNet-50 model checkpoint from [22], trained on the much larger Ego4D [42] (3,500 hours) rather than SSv2 (200 hours). As shown in Figure 5, HOLD-C outperforms R3M in both RLV tasks despite having been trained on less data and requiring no language descriptions. Like our method, R3M also requires sparse rewards to fully solve the tasks, and an intermediate goal for Drawer opening.

We also include a simple baseline of using the negative pixel-wise distance in image space $-||s - g||_2$ as a reward. While this baseline with sparse reward also learns Pushing faster than sparse reward alone, as shown in Figure 5, it still requires many more training samples than either HOLD-R or HOLD-C, and fails to reliably learn Drawer.



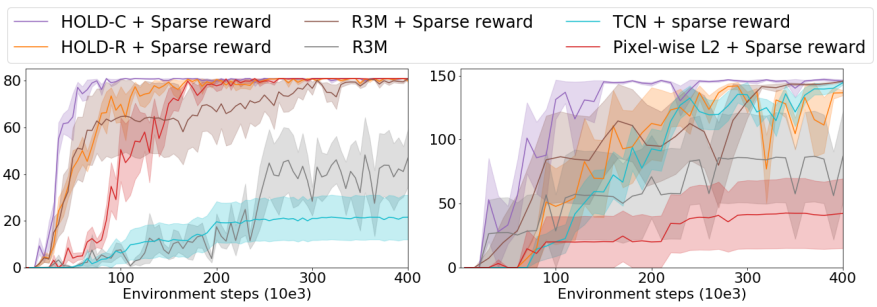
(a) Pushing (b) Drawer Opening (2 subtasks)

Fig. 3: Return on the RLV tasks (5 random seeds, with standard error).



(a) Close Drawer (b) Push Cup Forward (c) Turn Faucet Right

Fig. 4: Success rates on the DVD tasks (10 random seeds, with standard error). Our reward functions improve over sparse reward, and learn the Close Drawer task without using sparse reward.



(a) Pushing (b) Drawer Opening (2 subtasks)

Fig. 5: HOLD-C outperforms TCN and R3M rewards on both RLV tasks.

Parameter	HOLD-C	HOLD-R
Network	ResNet-50	ViViT
Epochs	100	20
Base learning rate	1e-4	0.1
Optimizer	Adam	Momentum
Batch size	8	64
Sequence length	32	
Margin (m)	0.2	
Pos. window	0.2s	
Neg. window	0.4s	

TABLE I: Training hyperparameters.

V. Conclusion

We have presented a method for learning goal image conditioned reward functions for robotic manipulation from unlabeled human videos, in a challenging setting which no prior work has addressed to our knowledge. Learning a prior for robot behavior from a dataset of human demonstrations without task labels requires generalization both across tasks and across a significant domain shift. While most accurate for short-horizon tasks, the distance functions we train produce useful rewards for visually different robot environments that are able to accelerate training over using sparse reward alone, and can be composed to perform more general multi-step manipulation tasks using subgoals. Finally, we have shown that for some tasks, the predicted rewards alone are sufficient to learn the task without any additional success signals.

Acknowledgements

This work was in part suppported by the Inria / NYU collaboration, the Louis Vuitton / ENS chair on artificial intelligence and the French government under management of Agence Nationale de la Recherche as part of the *Investissements d'avenir* program (PRAIRIE 3IA Institute) as well as under ANR 3IA MIAI@Grenoble Alpes (ANR-19-P3IA-0003). It was performed using HPC resources from GENCI–IDRIS (Grant 2022-AD011013362). Minttu Alakuijala was supported in part by a Google CIFRE PhD Fellowship. We would like to thank Elliot Chane-Sane for reviewing this manuscript.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.

[4] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, "Palm: Scaling language modeling with pathways," *arXiv preprint arXiv:2204.02311*, 2022.

[5] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6381–6385.

[6] 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.

[7] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 444–11 453.

[8] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural computation*, vol. 3, no. 1, pp. 88–97, 1991.

[9] J. Ho and S. Ermon, "Generative adversarial imitation learning," *Advances in Neural Information Processing Systems*, vol. 29, pp. 4565–4573, 2016.

[10] A. Kumar, J. Fu, M. Soh, G. Tucker, and S. Levine, "Stabilizing off-policy q-learning via bootstrapping error reduction," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[11] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," in *International Conference on Machine Learning*, 2019, pp. 2052–2062.

[12] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017, pp. 23–30.

[13] A. S. Chen, S. Nair, and C. Finn, "Learning generalizable robotic reward functions from "in-the-wild" human videos," in *RSS*, 2021.

[14] S. Tian, S. Nair, F. Ebert, S. Dasari, B. Eysenbach, C. Finn, and S. Levine, "Model-based visual planning with self-supervised functional distances," in *International Conference on Learning Representations*, 2021.

[15] K. Schmeckpeper, O. Rybkin, K. Daniilidis, S. Levine, and C. Finn, "Reinforcement learning with videos: Combining offline observations with interaction," in *Conference on Robot Learning*, 2020.

[16] K. Zakka, A. Zeng, P. Florence, J. Tompson, J. Bohg, and D. Dwibedi, "Xirl: Cross-embodiment inverse reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 537–546.

[17] L. Shao, T. Migimatsu, Q. Zhang, K. Yang, and J. Bohg, "Concept2robot: Learning manipulation concepts from instructions and human demonstrations," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1419–1434, 2021.

[18] A. Bonardi, S. James, and A. J. Davison, "Learning one-shot imitation from humans without humans," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3533–3539, 2020.

[19] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine, "One-shot imitation from observing humans via domain-adaptive meta-learning," in *RSS*, 2018.

[20] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 991–1002.

[21] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 1134–1141.

[22] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," in *Conference on Robot Learning*, 2022.

[23] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang, "Dexmv: Imitation learning for dexterous manipulation from human videos," in *ECCV 2022*. Springer, 2022, pp. 570–587.

[24] V. Petrík, M. Tapaswi, I. Laptev, and J. Sivic, "Learning object manipulation skills via approximate state estimation from real videos," in *Conference on Robot Learning*. PMLR, 2021, pp. 296–312.

[25] N. Das, S. Bechtle, T. Davchev, D. Jayaraman, A. Rai, and F. Meier, "Model-based inverse reinforcement learning from visual demonstrations," in *Conference on Robot Learning*, 2021, pp. 1930–1942.

[26] F. Torabi, G. Warnell, and P. Stone, "Behavioral cloning from observation," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 4950–4957.

[27] Y. Aytar, T. Pfaff, D. Budden, T. Paine, Z. Wang, and N. De Freitas, "Playing hard exploration games by watching youtube," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[28] I. Kostrikov, K. K. Agrawal, D. Dwibedi, S. Levine, and J. Tompson, "Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning," in *International Conference on Learning Representations*, 2019.

[29] Y. Wu, G. Tucker, and O. Nachum, "Behavior regularized offline reinforcement learning," *arXiv preprint arXiv:1911.11361*, 2019.

[30] Z. Wang, A. Novikov, K. Zolna, J. S. Merel, J. T. Springenberg, S. E. Reed, B. Shahriari, N. Siegel, C. Gulcehre, N. Heess *et al.*, "Critic regularized regression," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[31] X. B. Peng, A. Kumar, G. Zhang, and S. Levine, "Advantage-weighted regression: Simple and scalable off-policy reinforcement learning," *arXiv preprint arXiv:1910.00177*, 2019.

[32] H. Xiong, Q. Li, Y.-C. Chen, H. Bharadhwaj, S. Sinha, and A. Garg, "Learning by watching: Physical imitation of manipulation skills from human videos," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021, pp. 7827–7834.

[33] J. Li, T. Lu, X. Cao, Y. Cai, and S. Wang, "Meta-imitation learning by watching video demonstrations," in *International Conference on Learning Representations*, 2021.

[34] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, "Vip: Towards universal visual reward and representation via value-implicit pre-training," *arXiv preprint arXiv:2210.00030*, 2022.

[35] K. Hartikainen, X. Geng, T. Haarnoja, and S. Levine, "Dynamical distance learning for semi-supervised and unsupervised skill discovery," in *International Conference on Learning Representations*, 2020.

[36] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward transformations: Theory and application to reward shaping," in *International Conference on Machine Learning*, vol. 99, 1999, pp. 278–287.

[37] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on Robot Learning*. PMLR, 2020, pp. 1094–1100.

[38] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag *et al.*, "The "something something" video database for learning and evaluating visual common sense," in *Proceedings of the IEEE international Conference on Computer Vision*, 2017, pp. 5842–5850.

[39] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[40] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6836–6846.

[41] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1861–1870.

[42] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 995–19 012.