Translating Agent-Environment Interactions from Humans to Robots

Tanmay Shankar [1], Chaitanya Chawla [1,2], Almutwakel Hassan [1], and Jean Oh [1]

***Abstract*— Humans are remarkably adept at imitating other people performing tasks, afforded by their ability to abstract away irrelevant details and focus on the task strategy of the demonstrator. In this paper, we take steps towards enabling robots with this ability, and present a framework, *TransAct* to do so. *TransAct* first builds on prior skill learning work to learn temporally abstract representations of common agent-environment interactions in manipulation tasks, *e.g.,* a robot pouring from a cup. Given a human demonstration of an unseen unknown task, *TransAct* then translates the underlying sequence of interactions (*i.e.,* the human task strategy) to a robot learner. Through experiments on real-world human and robot datasets, we demonstrate *TransAct*'s ability to accurately represent diverse agent-environment interactions. Moreover, *TransAct* empowers robots to consume human task demonstrations and compose corresponding interactions with similar environmental effects to perform the tasks themselves in a zero shot manner, without access to paired demonstrations or dense annotations. We present visualizations of our results at https://sites.google.com/view/interaction-abstractions.**

I. Introduction

Consider an amateur cook learning a new dish by watching a chef demonstrate how to make a similar dish on YouTube; even amateurs can achieve excellent results doing so. This aptitude for "learning by imitation" is owed to the abstraction of human behaviours–including their own–and of the task at hand. People ignore irrelevancies (*e.g.,* differences in kitchens), and focus on patterns in the environmental changes needed (*e.g.,* steps of the cooking process), and skill-sequences that effect these environmental changes (*e.g.,* techniques of chefs, such as chopping or stirring skills). For example, when an amateur cook pours liquid into a pan from a cup, they adeptly ensure the cup progressively tilts at an increasing angle as liquid falls into the pan, by bending their wrists above the pan. The precise angle of the bottle and wrist matter less than the *pattern* of motion that *both* the bottle and wrist undergo; indeed, humans performing similar pouring motions across a variety of bottles and liquids.

This exemplifies how well people abstractly represent task strategies. Such task strategies capture the *how* the task is performed in addition to *what* task is performed, *i.e.,* they capture patterns of environmental change that occur during a task, and the sequence of skills they need to execute to accomplish the task at hand. The prospect of equipping *robots* with these abilities is enticing. Being able to transfer such abstract representations of task strategies would enable robots to consume human demonstrations, then execute their own corresponding skills that result in similar environmental changes, and even compose interactions they have only encountered individually to accomplish novel tasks.

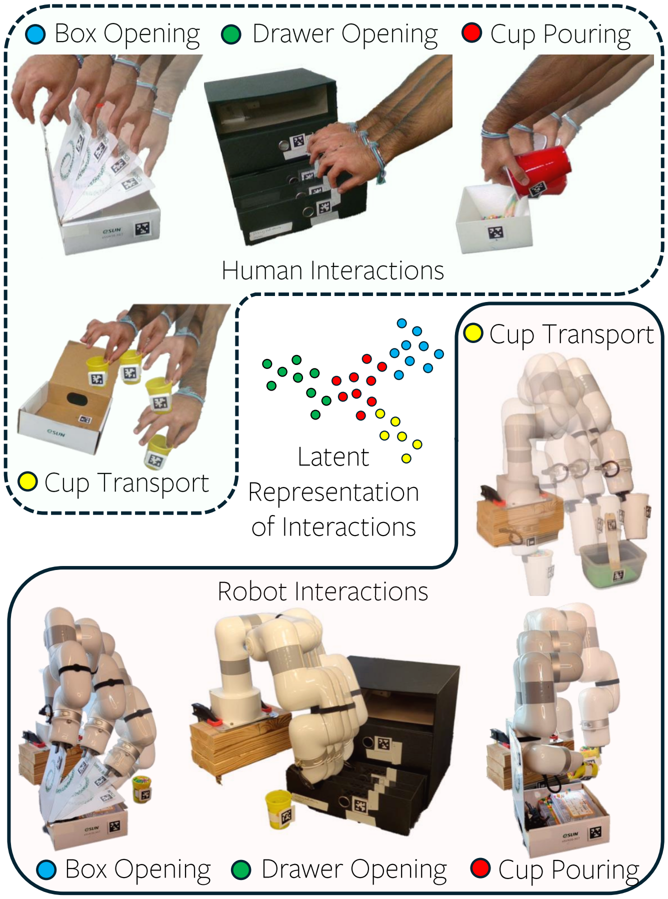

Fig. 1. Overview of interaction abstractions learnt by *TransAct*. We depict actual samples of the learnt representation space, visualized with corresponding trajectories from both human and real robot demonstrations. We depict 4 different interactions here, opening a box, opening a drawer, pouring from a cup, and transporting a cup.

In this paper, we take a step towards realizing this vision. Our primary contribution is *TransAct*, a framework that first learns temporally abstract representations of agent-environment interactions, then translates such interactions from human demonstrators to robot learners. *TransAct* has three important facets that enable it to do so; first, *TransAct* builds on prior robot skill learning work [1], and learns temporal abstractions of *interactions*, rather than modelling lower-level states or actions. We hope to provide a higher-level understanding of agent behaviors *and* their environmental effects, and therefore an understanding of which of their behaviors are needed to affect desired environmental changes. Second, *TransAct* models agent *and* environment

This work was supported by NSF IIS-2112633.

[1]The Robotics Institute, Carnegie Mellon University
[2]Technical University of Munich
Correspondence to: tanmay.shankar@gmail.com

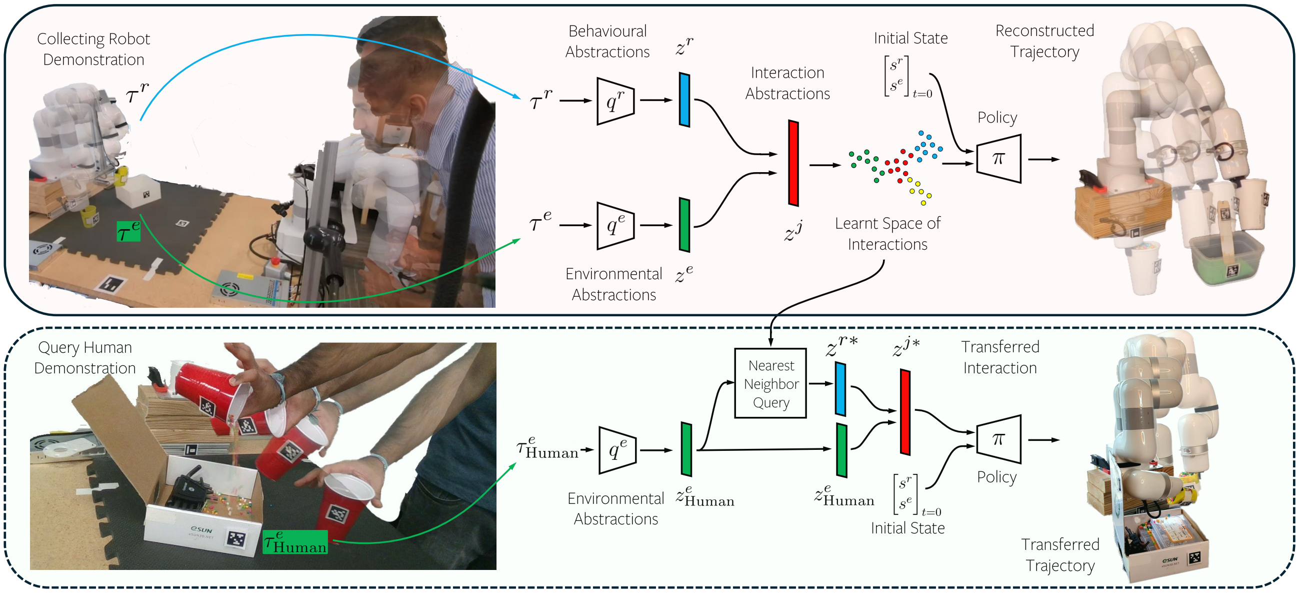

Fig. 2. Overview of *TransAct*. (a) our interaction abstraction learning approach (top) and (b) zero shot transfer approach (bottom). (a) A demonstrated robot trajectory τ^r and environment trajectory τ^e are encoded into behavioral and environmental abstractions z^r and z^e respectively, via their respective encoders q^r and q^e. z^r and z^e are combined into a joint interaction abstraction z^j. Robot policy π is then conditioned on z^j, in addition to state inputs s^r and s^e. (b) To transfer a given human query demonstration τ^e to the robot, we encode its environment abstraction, then retrieve the nearest neighbor robot interaction observed in the training dataset, and rollout the robot policy to retrieve the transferred trajectory.

trajectories for the *entire* duration of any given interaction (rather than just the start and goal), making it suitable for tasks where the full object trajectory is important, *e.g.,* pouring from a cup or stirring liquid in a cup. Third, *TransAct* facilitates transfer of these interactions across human demonstrators and robot learners. We consider in-domain transfer, *i.e.,* from human demonstrations collected in the same environment as the robot learners. Despite being restrictive, such in-domain transfer is still valuable. By transferring human interactions to robot interactions that lead to similar environmental effects as their human demonstrators, *TransAct* enables our learner robot to compose interactions it has only encountered individually to accomplish novel tasks specified by the human demonstrator even without access to paired demos, semantic or temporal segmentation annotations.

Our subsequent contributions are as follows. Second, we introduce a new training setting with several auxiliary objectives that imbibe *TransAct* with these traits. Third, we collect real-world human and robotic interaction datasets to evaluate our approach on. Finally, we demonstrate that *TransAct* accurately represents various diverse agent-environment interactions across both our real-world human and robot datasets, and existing simulated robot datasets; *TransAct* also translates real-world human demonstrations to our real-world robot, enabling it to perform these of complex novel tasks by composing interactions with similar environmental effects. We present visualizations of our results at https://sites.google.com/view/interaction-abstractions.

II. Related Work

1) Human to Robot Imitation Learning: Many works have addressed the human to robot imitation learning problem. [2], [3], [4], [5], [6], [7], [8], engineer mappings between demonstrators and robot state to facilitate imitation of human demonstrators. [9], [10], [11] have sought to learn such correspondences between human demonstrators and robots without manual specification, resorting to representation alignment machinery such as [12], or unsupervised domain adaptation machinery [13], [9]. These works have achieved remarkable success, by transferring individual states or actions across domains. However, with the exception of [9], these works lack a higher level understanding of the behaviors at hand [8], [7]. We argue that such higher-level abstractions are more transferrable across domains, as noted in [9]. [9] itself only transfers agent motion across domains. In contrast, our work aims to transfer higher-level abstractions of interactions across humans and robots.

2) Spatial and Temporal Abstractions of Behavior: The community has attempted to introduce higher level understanding in the form of abstractions. [14], [15], [16], [17] learn *state* abstractions that facilitate ignoring irrelevant components of environments, and making analogies across various environment and task instances [17]. [18], [19], [20], [21], [1], [22], [23] learn *temporal* abstractions of agent behavior, that facilitate reasoning over longer-term behaviors.

Despite making significant advances in building a high-level understanding of agent behavior, these works have significant pitfalls. Works that learn abstraction over agent behaviors [19], [21], [1], [22], [23], are often unaware of the patterns of environmental and object state change they induce (i.e., their effects). Conversely, most environmental state abstractions are unaware of the behaviors that caused them [14], [15], [16], [17], [24], [25]. These approaches therefore typically need to perform a search for an appropriate sequence of abstractions to solve the task - a difficult

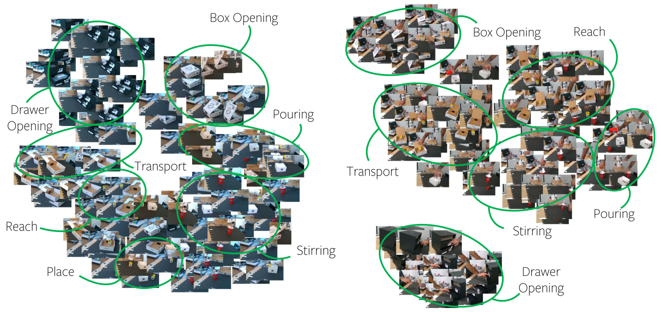

Fig. 3. Depiction of the learnt latent representation space of interactions from the real world robot and human datasets. Note the variety of interactions captured in the space, the clustering of similar interactions from similar tasks into similar parts of each space, and the local continuity of each space. For dynamic version of these spaces (and similar spaces for the other datasets), view https://sites.google.com/view/interaction-abstractions.

problem that often requires highly engineered heuristics (to plan with), or rewards (to learn from) to solve. In contrast, our work attempts to model the interaction between agent and environment; thereby enabling efficient retrieval of behaviors to cause a desired environmental effect; thus facilitating easy transfer of interactions across humans and robots.

3) Dynamics Aware Skill Learning: [18], [20], [26], [27] maintain notions of pre-conditions and effects alongside agent skills. These works typically only model the initial and final goal environment state. In contrast, our approach models the environment trajectory for the entire duration of the interaction, making them unsuitable for tasks where the object textittrajectory is important *e.g.,* pouring from a cup, opening a box or drawer, *etc.* Further, [18], [20], [26], [27] use the agent's own experience in a reinforcement learning context, and are therefore unsuitable for facilitating an agent imitating a human demonstrator, compared to our approach, which is directly trained on demonstration data.

III. Approach

A. Preliminaries – Learning Behavioral Abstractions

We first describe an important building block of our work–a behavioral abstraction framework. A behavioral abstraction, or skill, is a representation of an agent acting consistently for a temporally extended period, *e.g.,* a person stirring (a pot), or a person flipping an object such as a pancake, etc. We consider the behavioral abstraction framework of [1]; though any such framework could be used [19], [28], [22], [21], [29]. [1] learns behavioral abstractions, or skills, of agents from demonstrations in an unsupervised manner. Their method represents robot skills as continuous latent variables z^r (subscript r depicts the agent) and introduces a Temporal Variational Inference (TVI) to infer these skills or latent variables. Consider an agent state-action trajectory $\tau^r = \{s^r_1, a^r_1, ...s^r_{n-1}, a^r_{n-1}, s^r_n\}$, where s^r_t is agent state, a^r_t is the agent's action at time t, and n is trajectory length. TVI trains a variational encoder $q^r(z|\tau^r)$ that takes as input a agent trajectory τ^r and outputs a sequence of $k < n$ skill encodings $z^r = \{z^r_1, z^r_2, ...z^r_k\}$. TVI also trains a latent conditioned policy $\pi(a|s, z^r)$ that predicts the agent action a given the chosen skill encoding z, and. TVI optimizes q^r and π to maximize the likelihood of the *actions* observed in the trajectory τ^r. We direct the reader to [1] for a thorough description of their framework.

B. Learning Interaction Abstractions

1) Building Temporal Abstractions over Environment State: We can adapt the behavioral abstractions of [1] to learn equivalent temporal abstractions of environment state. Such abstractions could be used to specify patterns of object motion (more generally, changes in environmental state) that need to occur during a task, *e.g.,* , a bottle cap rotating and moving up away from the bottle, or a kettle being tilted downwards to pour from it. Consider a corresponding trajectory $\tau^e = \{s^e_1, a^e_1, ...s^e_{n-1}, a^e_{n-1}, s^e_n\}$ of *environment* state s^e over time. a^e_t represents the change in environmental state at a given timestep t, rather than a notion of agent action. We construct an equivalent *environmental* variational encoder $q^e(z|\tau^e)$, that predicts an equivalent sequence of latent encodings $z^e = \{z^e_1, z^e_2, ..., z^e_k\}$, that represent temporally abstract changes in environmental state.

2) Combining Behavioral and Environmental Abstractions into Interaction Abstractions: We now learn temporal abstractions of *interactions*, to build a high-level understanding of how agents interact with their environments. Given agent and environmental state trajectories τ^r & τ^e from a dataset, we encode these trajectories into their corresponding abstractions $\{z^r_1, z^r_2, ..., z^r_k\}$ & $\{z^e_1, z^e_2, ..., z^e_k\}$ via their respective encoders q^r and q^e. We then combine these behavioral and environmental abstractions by concatenating their encodings to form latent encodings of the interaction, $\{z^j_1, z^j_2, ..., z^j_k\}$, where *i.e.,* $z^j_k = \left[z^r_k \; z^e_k\right]$. The interaction abstractions $\{z^j\}$ specifies the sequence of environment state changes necessary for the task to be solved, as well as the sequence of skills the agent needs to execute to effect these changes. We inform the agent policy of this desired skill and desired pattern of environmental change by conditioning the policy π on z^j rather than on z^r alone as in [1]. Here, we can retrieve the agent's action by querying $\pi = \pi(a|s, \{z^j\}^k_{t=1})$. We depict this pictorially in fig. 2.

Such interaction abstractions (as opposed to behavioral abstractions alone) therefore provide a unified view of patterns of interactions across different agents in their environments. This in turn facilitates transfer across humans and robots, since we can retrieve skills that result in similar environmental changes across different agents.

C. Learning Representations of Interaction Abstractions

To learn representations of interactions z^j that facilitate downstream task transfer, there are 4 properties we desire of our representation, that TVI [1] does not afford.

1) Fidelity of underlying interactions: We would like the learnt representation to capture a given interaction with high fidelity, *i.e.,* accurately reconstruct the trajectories of both the agent and the environment from the representation. TVI optimizes for reconstruction of the agent's actions, but suffers when agent and environment *states* deviate from the demonstrations, due to the issue of compounding errors.

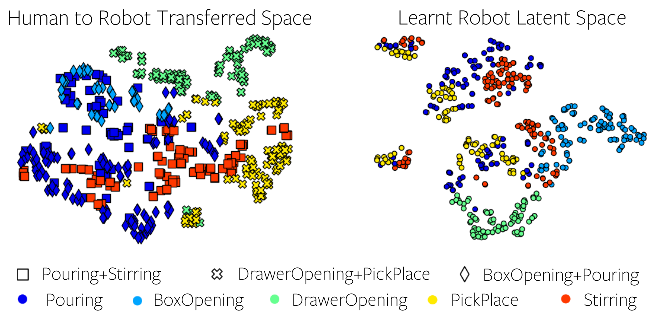

Fig. 4. Depiction of our transferred embedding space from humans $\mathcal{D}_{\text{HComp}}$ to robots $\mathcal{D}_{\text{R}}$ (left) and the robot latent space from $\mathcal{D}_{\text{R}}$ (right). In both figures, each symbol $\diamond, \square, \times, \circ$ represents an interaction; symbols $\diamond, \square, \times$ on the left represent the compositional query task a human demonstration is taken from. For both figures, the color the symbol represents the task (from $\mathcal{D}_{\text{R}}$) this interaction is from.

To mitigate this, we introduce an additional cumulative state reconstruction loss $\mathcal{L}_{\text{state}}$, encouraging the model to recover to the true states, from previous (potentially erroneous) state predictions made by the model itself. Given the model's predicted actions $\hat{a_t}$, we can integrate these actions over time to retrieve state predictions: $\hat{s_t} = s_{t=1} + \sum_{t'=1}^{t} \hat{a_{t'}}$. We then minimize the distance between the integrated state predictions $\hat{s_t}$ and the observed states s_t, for all timesteps t, *i.e.,* $\mathcal{L}_{\text{state}} = \sum_{t=1}^{T} \|\hat{s_t} - s_t\|_2^2$.

2) Architectural transfer from source to target: We facilitate transferring interactions from humans to robots via our architecture; by adopting a factored representation $z^j = \left[z^r \; z^e\right]$, where abstraction encoders q^e and q^r are independent of one another. Consider encoders q^r and q^e trained on a robot dataset $\mathcal{D}_{\text{R}}$. We reuse the learnt environment abstraction encoder q^e on arbitrarily complex query demonstrations from any other agent collected within the same environment in a zero-shot fashion, since the underlying environment trajectories themselves are from similar distributions.

3) Robustness and Smoothness of Representation: We facilitate easy retrieval and transfer of interactions across agents, by learning a locally continuous representation, *i.e.,* one that is robust to small perturbations in the input trajectories, such as small differences in environment trajectories across agents. We encourage this property by considering the Jacobian of the encoder $J_q = \frac{\partial q(z|\tau)}{\partial \tau}$. Past work has shown the benefit of regularizing the Jacobian of such encoder networks on the robustness of their representations [30] and local continuity in the representation [31]. We follow [30], and construct a regularization loss $\mathcal{L}_{\text{jac}}$ to regularize the Frobenius norm of the Jacobian, *i.e.,* $\mathcal{L}_{\text{jac}} = \|J_q\|_F^2$.

4) Contrastive Representations of Interactions: Intuitively, interactions from the same tasks, or interactions that result in similar object motions (or lack thereof) are easily retrieved if encoded into similar parts of the latent space, and further away from interactions from different tasks or inducing different object motions. We implement this by employing contrastive style losses [32], [33], [34] to place similar interactions close together *i.e.,* being *contractive* [31] w.r.t. similar interactions, and distinct skills farther apart, *i.e.,* being *discriminative* of distinct skills.

Consider a pair of interaction trajectory segments τ_m, τ_n. We introduce two contrastive losses between the representations of these interactions, z_m, z_n. The first loss, $\mathcal{L}_{\text{cont}}^{\text{task}}$, is based on whether these trajectories belong to the same task or not. For this loss, we define positive pairs via a membership function $\Delta_{m,n}^{\text{task}} = 1$ when τ_m, τ_n belong to the same task; negative pairs simply have $\Delta_{m,n}^{\text{task}} = 0$.

The second loss, $\mathcal{L}_{\text{cont}}^{\text{env}}$, is based on whether their environmental trajectories τ_m^e, τ_n^e are similar to each other or not. We use the the distance between representations of environment trajectories $\|z_m^e - z_n^e\|_2^2$ as a proxy of the distance between these trajectories; this choice is made reasonable because the Jacobian regularizer $\mathcal{L}_{\text{jac}}$ encourages local continuity in the representation. In principle, distances such as dynamic time warping distance between trajectories (as in [21]) could also be used. For this environment-trajectory based contrastive loss $\mathcal{L}_{\text{cont}}^{\text{env}}$, we define positive pairs via a membership function $\Delta_{m,n}^{\text{env}} = 1$ when the distance $\|z_m^e - z_n^e\|_2^2$ is below a threshold δ. Negative pairs have $\Delta_{m,n}^{\text{env}} = 0$.

For both $\mathcal{L}_{m,n}^{\text{task}}$ and $\mathcal{L}_{m,n}^{\text{env}}$, for a positive pair of interactions (*i.e.,* belonging to the same task, or having similar environmental trajectories), we pull their representations z_m, z_n closer together to within a threshold ϵ, by minimizing the positive loss $\mathcal{L}_{m,n}^{+} = \max(\epsilon, \|z_m - z_n\|_2^2)$. Similarly, for a negative pair of interactions, (*i.e.,* from different tasks, or having different environmental trajectories), we push their representations apart until they are at least ϵ apart, by minimizing the negative loss $\mathcal{L}_{m,n}^{-} = \max(0, \epsilon - \|z_m - z_n\|_2^2)$. For both $\mathcal{L}_{m,n}^{\text{task}}$ and $\mathcal{L}_{m,n}^{\text{env}}$, we compute the contrastive loss for every pair of trajectories in a batch $\mathcal{B}$,

$$\mathcal{L}_{\text{cont}}^{\{\text{env,task}\}} = \sum_{m,n \in \mathcal{B}} \left\{ \Delta_{m,n} \mathcal{L}_{m,n}^{+} + (1 - \Delta_{m,n}) \mathcal{L}_{m,n}^{-} \right\} \quad (1)$$

where $\Delta_{m,n}$ is the appropriate membership function from $\Delta_{m,n}^{\text{task}}, \Delta_{m,n}^{\text{env}}$. We can combine these losses into a single contrastive loss: $\mathcal{L}_{\text{cont}} = \mathcal{L}_{\text{cont}}^{\text{env}} + \mathcal{L}_{\text{cont}}^{\text{task}}$.

5) Final Objective: Our full objective supplements the TVI objective with our proposed auxiliary objectives: $\mathcal{L}_{\text{LIA}} = \mathcal{L}_{\text{TVI}} + \lambda_{\text{state}}.\mathcal{L}_{\text{state}} + \lambda_{\text{jac}}.\mathcal{L}_{\text{jac}} + \lambda_{\text{cont}}.\mathcal{L}_{\text{cont}}$, where λ's indicates the weights of our auxiliary losses.

D. Transferring Interactions from Humans to Robots

Given a human query demonstration τ_{Human} collected in the same environment as $\mathcal{D}_{\text{R}}$, we would like to transfer the interactions present in this demonstration to a robot, such that the environmental effects of the translated interactions are similar to those of the demonstration itself.

We operationalize this by first encoding the various interactions observed in the robot dataset $\mathcal{D}_{\text{R}}$ into a precomputed set of N interactions $\mathbb{Z} = \{z_1^j, z_2^j, ..., z_N^j\}$, using the encoders q^r and q^e trained on $\mathcal{D}_{\text{R}}$. We then encode the human query trajectory τ_{Human} into a sequence of query environment abstractions $\{z_k^e\}_{k=1}^K$, via the same q^e as above. We then retrieve the nearest neighbors of $\{z_k^e\}_{k=1}^K$, from the environmental component of the pre-computed latent space $\mathbb{Z}$, $\{z_k^{e*}\}_{k=1}^K$, and the corresponding *robot* behavioral abstractions $\{z_k^{r*}\}_{k=1}^K$ that would likely cause such environmental change. We then rollout our robot policy π conditioned on

TABLE I

State Reconstruction Error. Lower is better. * represents state errors computed over environment state alone. The full *TransAct* approach outperforms baseline approaches on reconstructing interactions across all datasets. *TransAct* also facilitates accurate reconstruction of interactions transferred from humans to robots comparable to approaches trained on the target domain data themselves.

Dataset	Baseline Approaches			*TransAct* + Ablations				
	VAE [35]	H-DMP [36]	TVI [1]	*TransAct* (Ours) Full	*TransAct* (Ours) No $\mathcal{L}_{\text{state}}$	*TransAct* (Ours) No $\mathcal{L}_{\text{jac}}$	*TransAct* (Ours) No $\mathcal{L}_{\text{cont}}$	*TransAct* (Ours) Un-Factored
Real Robot Data: $\mathcal{D}_{\text{R}}$ (Ours)	0.94	1.12	0.16	0.07	0.19	0.06	**0.04**	0.12
Human Data: $\mathcal{D}_{\text{H}}$ (Ours)	1.91	3.64	0.76	0.36	0.59	0.31	**0.28**	0.46
Human Data: $\mathcal{D}_{\text{HComp}}$ (Ours)	2.65	4.85	1.38	0.81	1.26	0.78	**0.57**	0.99
Human Data: $\mathcal{D}_{\text{H}}$ (Ours) *	0.83	1.74	0.26	0.12	0.34	0.11	**0.09**	0.17
H2R: $\mathcal{D}_{\text{H}} \rightarrow \mathcal{D}_{\text{R}}$ (Ours) *	3.13	3.75	2.98	**0.28**	0.59	1.04	1.19	2.83
Human Data: $\mathcal{D}_{\text{HComp}}$ (Ours) *	1.32	2.84	0.57	0.26	0.49	0.24	**0.21**	0.34
H2R: $\mathcal{D}_{\text{HComp}} \rightarrow \mathcal{D}_{\text{R}}$ (Ours) *	3.94	4.15	2.78	**0.40**	0.92	1.66	1.79	3.58
RoboTurk [37]	1.93	3.48	0.72	0.36	0.57	0.35	**0.31**	0.39
RoboMimic [38]	1.75	2.09	0.84	0.40	0.52	0.38	**0.32**	0.49
FrankaKitchen [39]	0.84	1.85	0.56	**0.22**	0.48	0.37	0.27	0.26
DAPG [40]	2.46	3.85	0.97	**0.46**	0.78	0.49	0.47	0.59
DexMV [41]	2.61	3.58	1.05	0.51	0.63	0.49	**0.46**	0.57

$\{z_k^{r*}, z_k^e\}_{k=1}^K$, to retrieve a robot trajectory that has the same environmental effect as the original query trajectory τ_{Human}. We depict this process pictorially in the bottom half of fig. 2.

IV. Datasets and Experimental Setup

We use several demonstration datasets to evaluate our approach on. Together, these datasets span human and robotic domains, real world and simulation, a variety of different objects (*e.g.,* boxes, cups, drawers, differently shaped pegs, etc.), and interactions (*e.g.,* including lifting, moving, releasing, pushing, rotating etc.).

1) Real-World Setup and Datasets: We collect a two small real-world datasets. The first is a robotic dataset $\mathcal{D}_{\text{R}}$ of a X-ARM Lite6 robot performing 5 different tasks, *Pouring from a cup*, *Stirring a cup*, *Box Opening*, *Drawer Opening*, and *Pick and Place*. We collect a parallel dataset, $\mathcal{D}_{\text{H}}$ of a human performing the same 5 individual tasks. We collect another human dataset, $\mathcal{D}_{\text{HComp}}$, performing 3 additional tasks composed from the individual tasks in $\mathcal{D}_{\text{H}}$. These tasks are *BoxOpening+Pouring*, *i.e.,* opening a box, then pouring beads from a cup into the box, *DrawerOpening+PickPlace*, *i.e.,* opening a drawer, then placing a cup into the open drawer, and finally *Pouring+Stirring*, *i.e.,* pouring beads into a cup, then stirring the poured beads.

We collect 10 robot and human demonstrations for each task, and 6 human demonstrations for each composed task in $\mathcal{D}_{\text{HComp}}$. We collect data on a "puppet" robot, that is teleoperated by kinesthetically controlling a "master" bot as in fig. 2. We record the 6 DoF joint state, and gripper state as the robot state, and treat joint-velocities and gripper opening and closing as the robot action. For each human demonstration, we employ the hand tracking from [42] to record a 25 dimensional joint state of the human hand (consisting of the base of the palm, index, thumb, and middle fingers). For the human and robot datasets, we place Apriltags [43] [44] [45] on each of the 2 objects involved a task for object-state estimation, and collect 6-D pose (position and orientation) of each object as the object state (also pictured in fig. 2).

2) Simulated Datasets: We also present results on the following publicly available simulated robot datasets — Roboturk [37] (Sawyer robot), RoboMimic [38] and FrankaKitchen [46], (Franka Panda), and DAPG [40] and DexMV [41] (Adroit hand). For each dataset, we use the robot joint states and gripper state as robot state, joint-velocities as robot actions, and the 6-D object pose of the primary object in the respective task as the object state. Note that we assume a fixed number of objects in each dataset, and concatenate object poses for each object as input to our model.

V. Experimental Results

We evaluate the ability of our proposed abstractions to *learn* and *transfer* interactions across humans and robots, and design a set of experiments to answer two questions. Firstly, how well can the learnt interaction abstractions accurately model interactions between agents and their environments? Secondly, how well do the learnt abstractions facilitate transferring task-strategies from humans to robots?

A. Modelling Interactions

1) Reconstructing Sequences of Interactions: We first evaluate how accurately our model can reconstruct sequences of agent-environment interactions. Not only does this implicitly require modelling individual interactions well, but can involve reconstructing a sequence of $15 - 20$ agent-environment interactions in some tasks.

a) Quantitative Reconstruction Measures: We encode and reconstruct each trajectory in each dataset, and compute the average mean-squared reconstruction error of both the agent and environment state, presented in table I. We compare our approach against several baselines based on a non-hierarchical VAE [35], a hierarchical primitive based approach that learns DMPs for each interaction segment (as opposed to our learnt representation) [36], and TVI [1], the skill learning approach [1]. As presented in table I, our approach achieves significantly lower reconstruction errors than the other baseline approaches, despite needing to satisfy additional constraints imposed by the auxiliary losses.

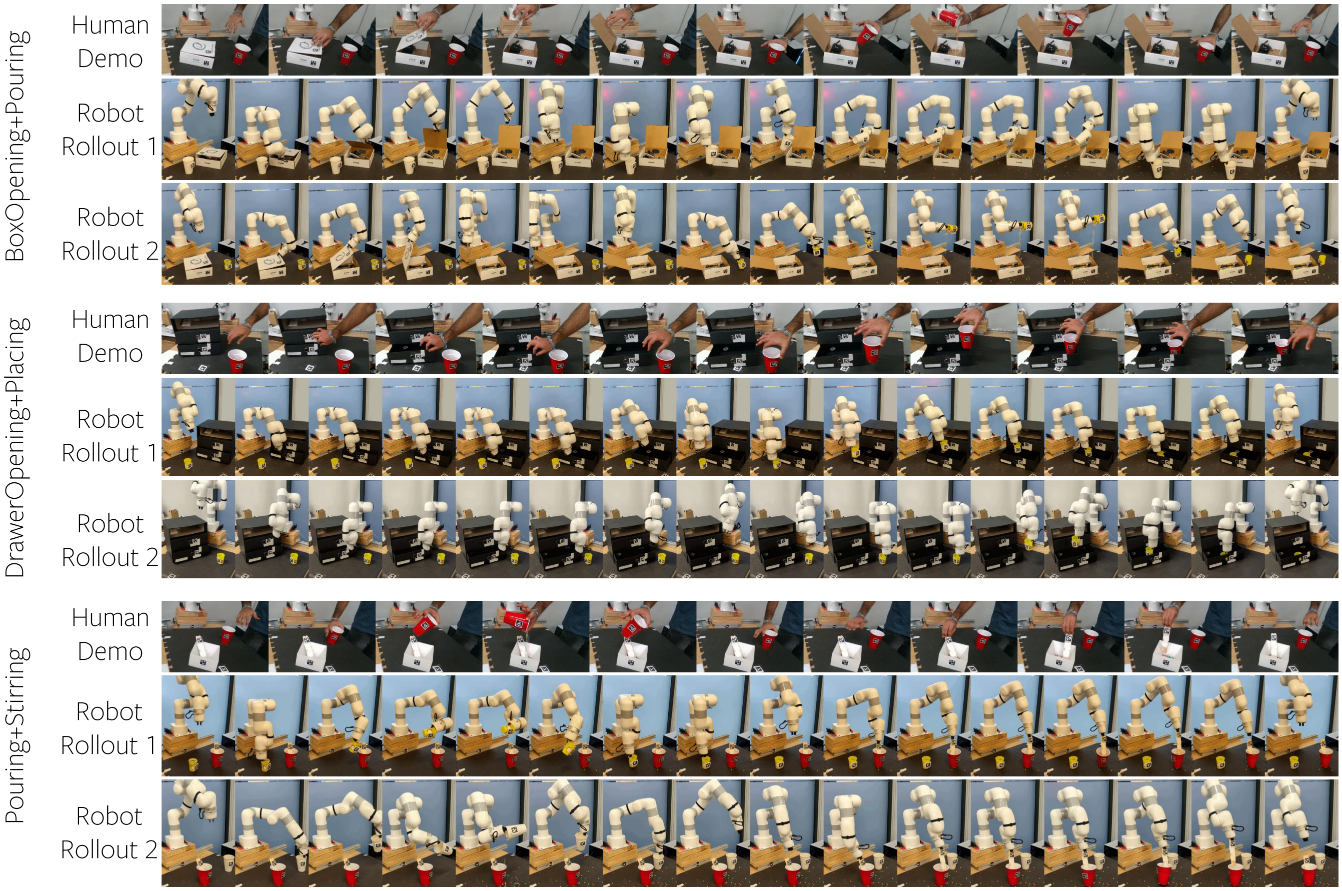

Fig. 5. Depiction of human query demonstrations for 3 different compositional tasks, as well as 2 transferred trajectories each executed on the robot. At the top, our approach translates the human sequence of reaching and opening a box to a sufficient angle to pour into, releasing the box, grasping and transporting a cup above the box, before pouring beads from the cup into the box. In the middle, we observe it is able to translate reaching to and opening the drawer, reaching and grasping the cup, transporting and placing the cup before finally returning. In the bottom, it similarly reaches for a cup, transports it above another container, pours beads, and then subsequently grasps a stirrer and stirs the container. Our approach is capable of doing this despite significant variation in the relative configuration of the objects.

b) Visualizing Interaction Reconstructions: We visualize reconstructions of individual interactions in fig. 1 for the human and robot datasets. As seen in fig. 1, the real robot is able to accurately reconstruct various interesting interactions observed in the demonstrations, such as pulling open a drawer, pushing the lid of a box open, and tilting an object to pour from it. We defer visualizing entire trajectories to our website and supplemental video due to space constraints.

2) Visualizing Space of Interactions: We also present 2D visualizations of the *space* of interactions for the real world robot and human datasets in fig. 3, produced by feeding a set of interactions $\{z_k^j\}_{k=1}^N$ to T-SNE [47]. We provide higher resolution dynamic visualizations of these spaces and other datasets on our website https://sites.google.com/view/interaction-abstractions. Qualitatively, we note the clustering of similar patterns of motion of objects from the same tasks into similar parts of the latent space, manually annotated for clarity. This clustering is afforded by our contrastive losses and jacobian regularizer. Our interaction abstraction space is able to capture a variety of different interactions, including picking objects, tilting an object (for pouring), pushing a box lid open, *etc.*

B. Facilitating Human to Robot Task Transfer

We now assess our framework's capability in transferring interactions between humans and robots.

1) Analysis of Human-to-Robot Transfer Reconstruction: We transfer query demonstrations from our human datasets $\mathcal{D}_{\mathrm{H}}$ and $\mathcal{D}_{\mathrm{HComp}}$ via the approach described in section III-D. We reiterate that the model is trained only on $\mathcal{D}_{\mathrm{R}}$, but evaluated on $\mathcal{D}_{\mathrm{H}}$ and $\mathcal{D}_{\mathrm{HComp}}$. Retrieving query trajectories from $\mathcal{D}_{\mathrm{HComp}}$ having trained our encoders on $\mathcal{D}_{\mathrm{R}}$ evaluates how well the model can compose seen interactions into novel and more complex sequences previously unseen by the robot.

We compute the reconstruction error between the *environmental* state of the rollouts and the query demonstrations (the agent states are incomparable in general); these results are presented in table I in rows H2R $\mathcal{D}_{\mathrm{R}} \rightarrow \mathcal{D}_{\mathrm{H}}$ and H2R $\mathcal{D}_{\mathrm{R}} \rightarrow \mathcal{D}_{\mathrm{HComp}}$, with the reconstruction error of a model trained on the human datasets $\mathcal{D}_{\mathrm{H}}$ and $\mathcal{D}_{\mathrm{HComp}}$ as baselines in rows $\mathcal{D}_{\mathrm{H}}$ * and $\mathcal{D}_{\mathrm{HComp}}$ *. In the transfer rows, H2R $\mathcal{D}_{\mathrm{R}} \rightarrow \mathcal{D}_{\mathrm{H}}$ and H2R $\mathcal{D}_{\mathrm{R}} \rightarrow \mathcal{D}_{\mathrm{HComp}}$, the transferred robot trajectories are able to achieve environment state reconstruction errors in similar ranges to those from the approach trained on $\mathcal{D}_{\mathrm{H}}$ and $\mathcal{D}_{\mathrm{HComp}}$ respectively. The baseline approaches and ablations of our approach without $\mathcal{L}_{\mathrm{cont}}$ and $\mathcal{L}_{\mathrm{jac}}$ all learn latent spaces where nearest neighbor retrieval from human to robot trajectories results in unrelated and inaccurate trajectories, leading to poor transfer reconstruction.

We visualize these transferred trajectories in fig. 5. Our approach composes sequences of robot-environment interactions to complete these novel compositional tasks specified by their human demonstrations. Note the ability of our model to follow the high level sequence of skills observed in the demonstrations. This is particularly notable despite our approach having access to only task-level labels, but not temporal segmentation labels, skill-level semantic labels, or other forms of supervision. Our approach is able to do this despite variation in the relative configuration of the objects from the human queries - indicating our approach goes beyond stitching together demonstrated trajectories. Our approach learns a spatially and temporally abstract representation of these interactions that offers a unified perspective of interactions across domains.

2) Analysis of Transferred Latent Spaces: We also present a visualization of the transferred latent space in fig. 4, akin to the visualization of the latent space in fig. 3. The transferred space shows that our approach can compose individual interactions to transfer human query demonstrations that consist of novel interaction sequences. The transferred space is also clustered similarly to the original robot latent space, despite being queried in a zero-shot fashion on *human* trajectories, further exemplifying the viability of our transfer approach. These results together indicate the ability of our approach to transfer human interactions to robot interactions that have a similar effect on their environments.

C. Ablation of various loss components

To determine the efficacy of our proposed auxiliary losses, we present an ablation study in the right half of table I, where our approach is trained without each one of the auxillary losses in different columns. As expected, removing the state-reconstruction loss $\mathcal{L}_{\text{state}}$ hurts the reconstruction performance for all datasets, as well as the human to robot transfer. Interestingly, removing the Jacobian and contrastive losses $\mathcal{L}_{\text{jac}}$ and $\mathcal{L}_{\text{cont}}$ *improve* state reconstruction on most individual datasets, but drastically hurts the downstream transfer performance. Here, the model is able to place more emphasis on reconstructing trajectories accurately, but fails to create a latent space conducive to transfer. Using a non-factored encoder model achieves similar reconstruction performance as our factored approach, but fails to aid transfer as ours does; this may be attributed to nearest neighbor retrieval of joint agent-environment abstractions z^j being less meaningful for different agents than retrieving pure environment abstractions.

VI. Limitations

In its current form, *TransAct* requires human demonstrations be collected in the same environment as the robot demonstrations. Relaxing these conditions would improve the applicability of our approach to human demonstrations in the wild, such as YouTube videos *etc.*We believe this could be achieved by constructing additional objectives to ground environmental states between human and robot domains, as in [12], [16]. *TransAct* is also currently unable to detect and recover from task failures from the stochastic nature of contact; this may be remedied by running a *closed* loop robot policy conditioned on learnt or transferred interactions.

VII. Conclusions and Future Work

Our primary contribution is our *TransAct* framework, which first learns abstract representations of agent-environment interactions, then translates such interactions from human demonstrators to robot learners. Despite the limitations listed above, *TransAct* serves as a valuable tool to enable learner robots to consume human task demonstrations, and compose interactions with similar environmental effects encountered individually to accomplish these novel tasks specified by a human demonstrator. *TransAct*'s ability to model both the agent *and* environment for the entire duration of their interaction is powerful. This is particularly valuable when a learner robot has some primitive skills, but learning how to compose such skills to solve more complex tasks is expensive, a scenario that is increasingly prevalent in the robotics community. We hope that our work on enabling the transfer of task strategies (the *how* and the *what* of tasks) from humans to robots facilitates further research in this domain.

Acknowledgements

We would like to extend special thanks to Eliot Xing and Peter Schaldenbrand for their help in setting up the robot infrastructure and running robot experiments.

References

[1] T. Shankar and A. Gupta, “Learning robot skills with temporal variational inference,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 13–18 Jul 2020, pp. 8624–8633. [Online]. Available: https://proceedings.mlr.press/v119/shankar20b.html

[2] A. Sivakumar, K. Shaw, and D. Pathak, “Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube,” *RSS*, 2022.

[3] S. P. Arunachalam, S. Silwal, B. Evans, and L. Pinto, “Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation,” *arXiv preprint arXiv:2203.13251*, 2022.

[4] J. Ye, J. Wang, B. Huang, Y. Qin, and X. Wang, “Learning continuous grasping function with a dexterous hand from human demonstrations,” 2022.

[5] 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,” 2022.

[6] Y.-H. Wu, J. Wang, and X. Wang, “Learning generalizable dexterous manipulation from human grasp affordance,” 2022.

[7] F. O. H. to Multiple Hands: Imitation Learning for Dexterous Manipulation from Single-Camera Teleoperation, “Qin, yuzhe and su, hao and wang, xiaolong,” 2022.

[8] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox, “Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system,” in *Robotics: Science and Systems*, 2023.

[9] T. Shankar, Y. Lin, A. Rajeswaran, V. Kumar, S. Anderson, and J. Oh, “Translating robot skills: Learning unsupervised skill correspondences across robots,” in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 19 626–19 644. [Online]. Available: https://proceedings.mlr.press/v162/shankar22a.html

[10] L. M. Smith, N. Dhawan, M. Zhang, P. Abbeel, and S. Levine, "AVID: learning multi-stage tasks via pixel-level translation of human videos," *CoRR*, vol. abs/1912.04443, 2019. [Online]. Available: http://arxiv.org/abs/1912.04443

[11] H. Bharadhwaj, A. Gupta, V. Kumar, and S. Tulsiani, "Towards generalizable zero-shot manipulation via translating human interaction plans," 2023.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[13] C. Zhou, X. Ma, D. Wang, and G. Neubig, "Density matching for bilingual word embedding," in *Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, Minneapolis, USA, June 2019. [Online]. Available: https://arxiv.org/abs/1904.02343

[14] C. Gelada, S. Kumar, J. Buckman, O. Nachum, and M. G. Bellemare, "DeepMDP: Learning continuous latent space models for representation learning," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 2170–2179. [Online]. Available: https://proceedings.mlr.press/v97/gelada19a.html

[15] L. Li, T. J. Walsh, and M. L. Littman, "Towards a unified theory of state abstraction for mdps," in *International Symposium on Artificial Intelligence and Mathematics, AI&Math 2006, Fort Lauderdale, Florida, USA, January 4-6, 2006*, 2006. [Online]. Available: http://anytime.cs.umass.edu/aimath06/proceedings/P21.pdf

[16] Q. Zhang, T. Xiao, A. A. Efros, L. Pinto, and X. Wang, "Learning cross-domain correspondence for control with dynamics cycle-consistency," 2021. [Online]. Available: https://openreview.net/forum?id=QIRlze3I6hX

[17] P. Hansen-Estruch, A. Zhang, A. Nair, P. Yin, and S. Levine, "Bisimulation makes analogies in goal-conditioned reinforcement learning," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 8407–8426. [Online]. Available: https://proceedings.mlr.press/v162/hansen-estruch22a.html

[18] R. S. Sutton, D. Precup, and S. Singh, "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning," *Artificial intelligence*, 1999.

[19] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine, "Diversity is all you need: Learning skills without a reward function," 2019. [Online]. Available: https://openreview.net/forum?id=SJx63jRqFm

[20] A. Sharma, S. Gu, S. Levine, V. Kumar, and K. Hausman, "Dynamics-aware unsupervised discovery of skills," 2020. [Online]. Available: https://openreview.net/forum?id=HJgLZR4KvH

[21] T. Shankar, S. Tulsiani, L. Pinto, and A. Gupta, "Discovering motor programs by recomposing demonstrations," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=rkgHY0NYwr

[22] S. Krishnan, R. Fox, I. Stoica, and K. Goldberg, "Ddco: Discovery of deep continuous options for robot learning from demonstrations," *arXiv preprint arXiv:1710.05421*, 2017.

[23] R. Fox, S. Krishnan, I. Stoica, and K. Goldberg, "Multi-level discovery of deep options," *arXiv preprint arXiv:1703.08294*, 2017.

[24] T. Kim, S. Ahn, and Y. Bengio, "Variational temporal abstraction," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 11 566–11 575. [Online]. Available: http://papers.nips.cc/paper/9332-variational-temporal-abstraction.pdf

[25] K. Gregor, G. Papamakarios, F. Besse, L. Buesing, and T. Weber, "Temporal difference variational auto-encoder," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=S1x4ghC9tQ

[26] B. Freed, S. Venkatraman, G. A. Sartoretti, J. Schneider, and H. Choset, "Learning temporally AbstractWorld models without online experimentation," in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 23–29 Jul 2023, pp. 10 338–10 356. [Online]. Available: https://proceedings.mlr.press/v202/freed23a.html

[27] S. Park, O. Rybkin, and S. Levine, "Metra: Scalable unsupervised rl with metric-aware abstraction," 2023.

[28] S. Krishnan, A. Garg, S. Patil, C. Lea, G. Hager, P. Abbeel, and K. Goldberg, "Transition state clustering: Unsupervised surgical trajectory segmentation for robot learning," in *RR*, 2018.

[29] T. Kipf, Y. Li, H. Dai, V. Zambaldi, A. Sanchez-Gonzalez, E. Grefenstette, P. Kohli, and P. Battaglia, "Compile: Compositional imitation learning and execution," in *ICML*, 2019.

[30] J. Hoffman, D. A. Roberts, and S. Yaida, "Robust learning with jacobian regularization," 2020. [Online]. Available: https://openreview.net/forum?id=ryl-RTEYvB

[31] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: explicit invariance during feature extraction," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ser. ICML'11. Madison, WI, USA: Omnipress, 2011, p. 833–840.

[32] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 1735–1742.

[33] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[34] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/6b180037abbebea991d8b1232f8a8ca9-Paper.pdf

[35] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.

[36] S. Niekum, S. Osentoski, G. Konidaris, and A. G. Barto, "Learning and generalization of complex tasks from unstructured demonstrations." IEEE, 2012.

[37] A. Mandlekar, Y. Zhu, A. Garg, J. Booher, M. Spero, A. Tung, J. Gao, J. Emmons, A. Gupta, E. Orbay, S. Savarese, and L. Fei-Fei, "Roboturk: A crowdsourcing platform for robotic skill learning through imitation," in *Conference on Robot Learning*, 2018.

[38] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, "What matters in learning from offline human demonstrations for robot manipulation," in *arXiv preprint arXiv:2108.03298*, 2021.

[39] C. Lynch, M. Khansari, T. Xiao, V. Kumar, J. Tompson, S. Levine, and P. Sermanet, "Learning latent plans from play," *arXiv preprint arXiv:1903.01973*, 2019.

[40] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations," in *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

[41] 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," 2021.

[42] M. Contributors, "Openmmlab pose estimation toolbox and benchmark," https://github.com/open-mmlab/mmpose, 2020.

[43] D. Malyuta, C. Brommer, D. Hentzen, T. Stastny, R. Siegwart, and R. Brockers, "Long-duration fully autonomous operation of rotorcraft unmanned aerial systems for remote-sensing data acquisition," *Journal of Field Robotics*, p. arXiv:1908.06381, Aug. 2019. [Online]. Available: https://doi.org/10.1002/rob.21898

[44] C. Brommer, D. Malyuta, D. Hentzen, and R. Brockers, "Long-duration autonomy for small rotorcraft UAS including recharging," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, oct 2018, p. arXiv:1810.05683. [Online]. Available: https://doi.org/10.1109/iros.2018.8594111

[45] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, oct 2016, pp. 4193–4198.

[46] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman, "Relay policy learning: Solving long horizon tasks via imitation and reinforcement learning," *Conference on Robot Learning (CoRL)*, 2019.

[47] L. van der Maaten and G. Hinton, "Visualizing high-dimensional data using t-sne," 2008.