Video-Guided Skill Discovery

Manan Tomar [*1] **Dibya Ghosh** [*2] **Vivek Myers** [*2] **Anca Dragan** [2] **Matthew E. Taylor** [1]
Philip Bachman [3] **Sergey Levine** [2]

Abstract

We study how embodied agents can use passive data, such as videos, to guide the discovery of useful and diverse skills. Existing datasets have the potential to be an abundant and rich source of examples for robot learning, revealing not only *what* tasks to do, but also *how* to achieve them. Without structural priors, existing approaches to skill discovery are often underspecified and ineffective in real-world, high-DoF settings. Our approach uses the temporal information in videos to learn structured representations of the world that can then be used to create shaped rewards for efficiently learning from open-ended play and fine-tuning to target tasks. We demonstrate the ability to effectively learn skills by leveraging action-free video data in a kitchen manipulation setting and on synthetic control tasks.

1. Introduction

Internet video has the potential to be an effective and low-cost source of data for robot learning. When learning from such prior data, an agent must be able to adapt the extracted knowledge so that it is useful in the agent's own observation and action spaces. One approach to ensuring that such knowledge is useful is by learning skills that realize the behavior observed in the prior data. Learning such skills provides an opportunity to replicate behavior present in diverse internet videos in any given embodied agent's environment without any extrinsic reward, which can then be used for solving any downstream tasks more quickly. How should an agent go about learning such skills from the prior data without any extrinsic reward?

Most prior work on learning from video has learned representations which are directly evaluated through downstream TD-learning or imitation learning, assuming access to expert actions or environment rewards. However, the notion of learning skills and representations in an unsupervised manner when given access to an embodied agent has been largely unexplored. On the other hand, most prior methods that discover unsupervised skills do so using an intrinsic reward that maximizes mutual information of skills, but leads to collapse in many scenarios. We view learning from video data as a way of enforcing a strong prior that prefers to visit certain states or perform certain behaviors. When acting in embodied settings, we propose optimizing for skill representations that align with the behavior seen in the prior data.

This paper discusses how such a framework can used to learn behaviors from video data. Our approach encodes skills from prior data and learns a value function corresponding to the behavior described by the encoded skill. We then define an intrinsic reward for an embodied agent using the value function and skill encoder trained on prior data. Finally, we adapt the value model on the data collected by the embodied agent. Optimizing the intrinsic reward and adapting the value model both help bridge the gap between desired behavior (i.e. that in the prior data) and realized behavior (i.e. that in the embodied agent's environment). An outline of our approach is presented in Figure 1.

Overall, **1)** we introduce a framework for learning from video data when an agent is embodied with varying observation/action spaces without extrinsic reward, **2)** show how skill discovery can be made feasible by providing structure through preferences/intentions in the prior video data and **3)** show preliminary results on learning skills and representations from prior data across a PointMass and a pixel-based simulated Kitchen environment.

2. Related Work

Unsupervised Skill Discovery. Prior work on skill discovery often operates in an unsupervised fashion with a learned skill-conditioned policy (Eysenbach et al., 2018a; Sharma et al., 2019; Laskin et al., 2022). Generally, the skill is a continuous random variable which is used to sample actions

[*]Equal contribution [1]University of Alberta [2]University of Berkeley [3]MSR Montreal. Correspondence to: Manan Tomar <manan.tomar@gmail.com>, Dibya Ghosh <dibya.ghosh@berkeley.edu>, Vivek Myers <vmyers@berkeley.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

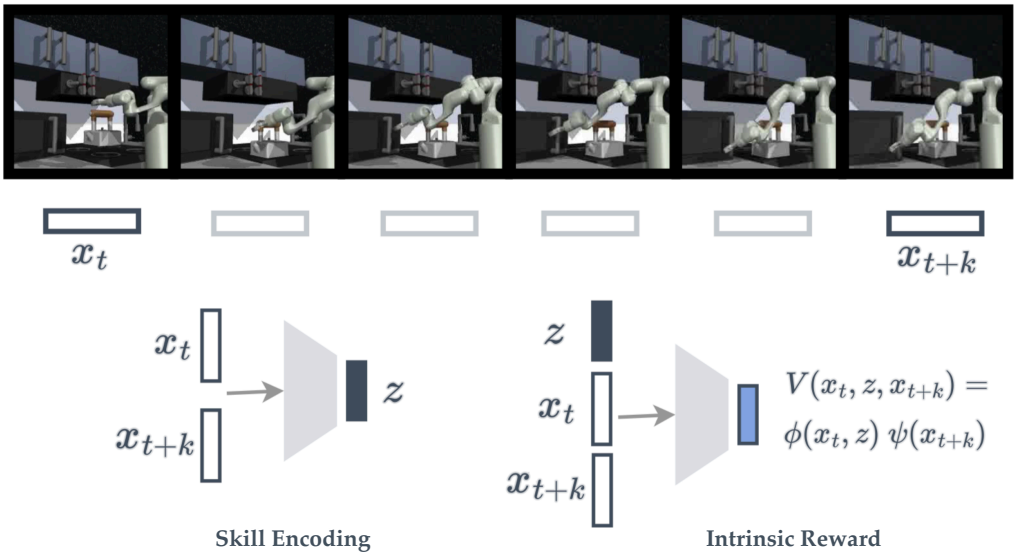


Figure 1. **Schematic for Learning Skills from Video**. We learn a discrete skill encoding given states in the video, and use it to condition a behavior value model. The embodied agent then uses the value model as an intrinsic reward function to learn video skills in it's own environment. The behavior value model is adapted using the data collected by the embodied agent.

from the policy. The reward for the policy learning is usually some form of discriminator output that incentivizes the agent to produce skills that are more discernible. Most methods falling in this category lead to objectives that encourage covering all possible states, which has been shown to be quite unstable in practice.

State Distribution Matching. A separate line of work mimics state distributions that correspond to expert data. These works estimate a state density model and reward the policy for matching this density with that of the prior data. GAIL (Ho and Ermon, 2016) uses state-action pairs from an expert to learn a discriminator function while using its score to reward the policy. Our approach is similar insofar as we use a discriminator to define the intrinsic reward, i.e., the cosine similarity. However, the discriminator is defined over skills (from the video data) instead of being defined over state occupancy. Furthermore, we also do not use any action information from the video data. SMM (Lee et al., 2019) rewards the policy for matching an estimated state density with a target state distribution. The main caveat is that SMM assumes a given target distribution, as well as that estimating state densities can be arbitrarily complex for diverse video data.

GAIL from observations and similar approaches (Torabi et al., 2019b;d;a;c; Peng et al., 2022) learn an observation-only discriminator to use as an intrinsic reward. Unlike GAIL-based approaches, our algorithm does not rely on potentially unstable adversarial training and obtains skill-conditioned policies by align inferred skills with state transitions seen in the data. Furthermore, we utilize a contrastive learning objective as opposed to a MSE-based formulations in prior work. BCO (Torabi et al., 2018a) infers actions using an inverse model and then runs behavior cloning over them. However, it does not account for the distribution mismatch between the state-action pairs produced by the expert and the agent.

Learning from Video. Recent work has introduced training representations useful for motor control on large datasets. These representations are trained to be temporally consistent by defining the current and future states in a video trajectory as positive contrastive pairs. Our representation learning objective is also defined around temporal consistency and achieves this through some form of contrastive learning. However, crucially, we optimize for learning minimal information containing skills (through discrete skills) which puts emphasis on learning representations of behaviors instead of time-consistent state representations, as in VIP (Ma et al., 2022) and R3M (Nair et al., 2022). Furthermore, once a representation is learnt, we use that to discover skills that are seen in the video data, whereas VIP/R3M consider learning the representation as the end goal and so directly evaluate its quality with behavior cloning. Recently, the ICVF (Ghosh et al., 2023) model has been used to learn representations by capturing intent-conditioned value functions. Besides not learning skills in an unsupervised setting, ICVFs also do not condition on a latent intent and instead uses state-level goals as a proxy for intents.

Algorithm 1 Video-guided Skill Discovery

Input: Prior video data $\mathcal{D}_{\text{prior}}$
State encoder networks ϕ and ψ
Skill encoder and decoder q_{enc}, q_{dec}
Sample skill z from a categorical distribution before every episode
for $m = 1, \ldots, M$ **do**
  Take action $a_t \sim \pi(\cdot \mid x_t, z)$, observe x_{t+1}, and add to the replay buffer $\mathcal{D}_{\text{online}}$
  for N updates **do**
    Sample prior data batch as $\{x_t, x_{t+k}\}_{j=1}^{B}$ from $\mathcal{D}_{\text{prior}}$ ▷ update on prior/video data
    Shuffle x_{t+k} to get negatives x_{t+k}^{neg}
    Cluster skills in prior data by encoding and decoding $z_{\text{enc}} = q_{\text{enc}}(x_t), z_{\text{dec}} = q_{\text{dec}}(x_{t+k})$
    Optimize prior data value loss $\mathcal{L}_{\text{value}} = \cos(\phi(x_t, z_{\text{enc}}),\ \psi(x_{t+k})) - \cos(\phi(x_t, z_{\text{enc}}),\ \psi(x_{t+k}^{\text{neg}}))$
    Optimize skill clustering loss $\mathcal{L}_{\text{skill}} = \text{infoNCE}(z_{\text{enc}}, z_{\text{dec}})$
    Sample agent data batch as $\{x_t, x_{t+k}\}_{j=1}^{B}$ from $\mathcal{D}_{\text{online}}$ ▷ update on agent data
    Shuffle x_{t+k} to get negatives x_{t+k}^{neg}
    Optimize agent data value loss $\mathcal{L}_{\text{value}} = -\cos(\phi(x_t),\ \psi(x_{t+k}^{\text{neg}}))$ ▷ update on agent policy
    Run DDPG update on $\pi(a|x_t, z)$ for reward $r = \phi(x_t, z)^{\text{T}}\psi(x_{t+k}) + ||z - q_{\text{enc}}(x_t)||^2$
  end for
end for

3. Discovering Skills with Video Guidance

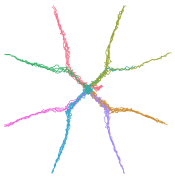
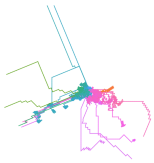
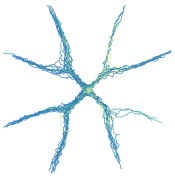
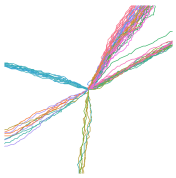
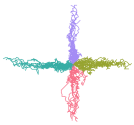
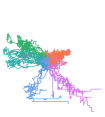
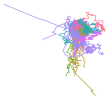
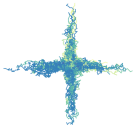
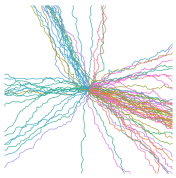
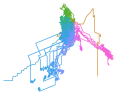
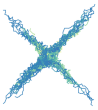
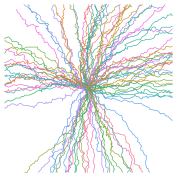
We are interested in controlling an agent in an environment provided access to a video dataset, with the goal of acquiring useful skills for performing downstream tasks. Intuitively, for such an agent, learning decomposes into three phases: first, a pretraining phase during which the agent can train on this prior data; second, an embodied skill acquisition phase, in which the agent is given unsupervised access to the environment; and finally, a downstream finetuning phase, where an extrinsic reward is specified which the agent must learn to maximize as quickly as possible.

Problem Setup: Formally, the agent acts in an Markov control process $\mathcal{M} := (\mathcal{S}, \mathcal{A}, P)$ with state space $\mathcal{S}$, action space $\mathcal{A}$, and transition kernel $P(s'|s, a)$. During the downstream finetuning phase, the agent will be asked to maximize expected discounted return for a task reward $r(s)$ coming from some distribution $p(r)$. The agent does not know the task distribution; rather, it is provided with a dataset $\mathcal{D} = \{\tau_i\}_{i=1}^{n}$ of video state trajectories $\tau_i := (x_0, x_1, \ldots, x_T)$ corresponding to acting optimally according to rewards from this distribution. For simplicity of exposition, we notate the video as coming from the same observation space of the agent from hereon, but in our experiments, we also consider settings where image videos are used to guide a robot with visual and proprioceptory inputs.

We motivate our approach by the following intuition: prior video data not only specifies *what* is interesting to achieve, but also *how* such behaviors might be achieved. Correspondingly, the high-level objective of our agent will be to generate useful tasks (in some appropriately defined task space), i.e. the "what to achieve" part, and to model the value of achieving said tasks, i.e. the "how to achieve" part. We capture the notion of tasks by encoding a skill based on current and future states $z = \text{enc}(x_t, x_{t+k})$. We capture the notion of value by defining a linear function composing the current and future state representations, conditioned on a task z, $\phi(x_t, z)^{\text{T}}\psi(x_{t+k})$. The overall recipe then looks like the following: **1)** Capture prior data in representations of current and future states $\phi(x_t)$ and $\phi(x_{t+k})$, so as to encode the prior behavior into a skill variable z, **2)** Use the encoded z and the corresponding value model to reward the policy of the embodied agent for going to future states that are aligned with the behavior seen in the prior data, **3)** adapt ϕ, ψ and z according to what trajectories are actually seen by the embodied agent. We now describe each of these steps in more detail.

Learning Representations from Prior Data. We learn state representations by encoding any current state x_t and future state x_{t+k} into representations $\phi(x_t)$ and $\psi(x_{t+k})$. The skill variable is separately encoded using the current and future states, $z = \text{enc}(x_t)$, where enc() denotes a simple neural network transformation. The skill variable gives us a notion of what behavior was followed from the current state to get to a future state. We learn the skill encoding by decoding a skill from given future states (i.e. which skills are most likely given a batch of future states), and matching those with the encoded skill. We use the encoded skill to get an estimate of the value of a followed behavior $V = \phi(x_t, z)^{\text{T}}\psi(x_{t+k})$. This value function is trained using contrastive learning, where actual future states form the positive samples, while random states (those belonging to other trajectories) form the negative samples. These two objectives lead to training a skill model and skill conditioned

Table 1. Comparison with Baselines on Point Mass

Prior Data	**Ours**	GCRL	DISCERN	GAIfO	DIAYN
					
					
					

value model, with only access to prior data. Next we discuss how these two models can be used to learn skills and how should these be adapted for the embodied agent.

Learning Skills for Embodied Agent. Having trained the value model and the skill encoder, we can look to learn policies that match certain skills. We do this by simply defining an intrinsic reward for a skill conditioned policy $\pi(a|x, z)$ which follows directly from the value model, i.e. $r_{\text{value}} = V = \phi(x_t, z)^{\text{T}}\psi(x_{t+k})$, where x_t and x_{t+k} are states seen by the embodied agent. Furthermore, to match the behavior of the skill-conditioned policy to that of the skill model trained on the prior data, we add a separate skill matching term to the reward, i.e. $r_{\text{skill}} = ||z - \text{enc}(x_t, x_{t+k})||^2$. The overall reward therefore is $r(s, z) = r_{\text{value}} + r_{\text{skill}}$. The first term guides the agent in learning how to perform a skill while the second term encourages the agent to map its skills, i.e. those that it uses to collect data, to that of the skill encoder trained on the video data.

Adapting the Representations with Agent Data. Once the embodied agent interacts with the environment in an unsupervised manner, it is natural to have a notion for adapting the pretrained value model given the agent's interaction. We do this by training the value model only on the negatives of the contrastive loss, $V = \phi(x_t, z)^{\text{T}}\ \psi(x'_{t+k})$, where x'_{t+k} denote the negative pairs, i.e. future states that are not observed when following skill z from state x_t. Note that the negative pairs of the value model help adapt the representations that comprise the value model, while the positve pairs of the value model help adapt the skill policy through the value component of the intrinsic reward r_{value}. Also note that adversarial approaches use a similar looking objective but regard all current and future state pairs as negatives. On the other hand, we remain "weakly-adversarial" since the negatives are defined only by the future states that the agent did not visit in its current trajectory.

We describe the complete working of our method in Algorithm 1, denoting each of the three major steps from above as "update on prior/video data", "update on agent policy," and "update on agent data" respectively.

4. Experiments

In this section, we test our approach for two primary requirements. The first is that of reaching *multiple* goals through the learnt skills. If the video data contains diverse or multi-goal reaching behavior, we should be able to capture all such possible behaviors. Secondly, we test whether our method is able to learn a notion of *skills/behaviors* present in the video data, and does not simply learn to capture state occupancy. This is important since as the video data gets more complex, any method that learns to capture state occupancy will likely suffer from the gap between the agent's current environment and the video data. This is to say that the intrinsic reward we described above should have a notion of what task or behavior was followed.

We choose two different domains to test how well our method fits the above two requirements. Firstly, we have a 2D PointMass with prior data coming in the form of state trajectories. Secondly, we have a pixel-based Kitchen envi-

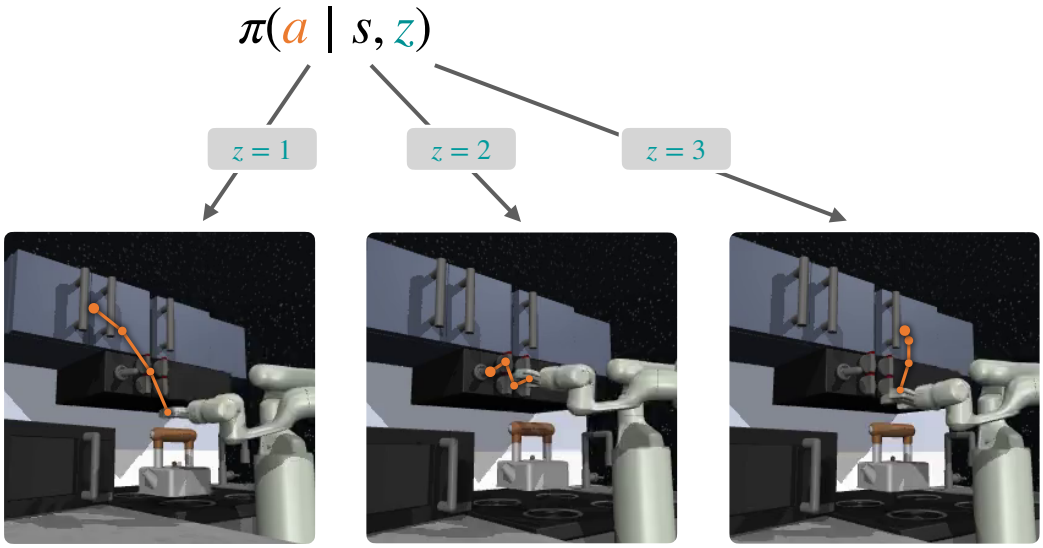


Figure 2. Trajectories from our approach with different sampled skills. We use video of the arm performing different tasks in the kitchen setting to learn a policy conditioned on skills z. We visualize three rollouts of different skills from our trained policy, manipulating the left door (left), light switch (center), and right door (right).

ronment (Gupta et al., 2019) where prior data comes in the form of videos of expert-like skills.

Baselines. We largely compare with two categories of baseline methods in the PointMass setting—skill discovery methods and state distribution matching methods. For skill discovery, we include DIYAN, and for state distribution matching we include DISCERN and GAIL from observations (GAIfO). We also consider a GCRL baseline which simply samples the goals from the prior data.

GCRL: GCRL performs goal-conditioned RL on a mixture of agent and prior data with hindsight experience replay (Andrychowicz et al., 2017).

DISCERN: DISCERN adapts GCRL to additionally use a contrastive classifier of goal attainment as a reward term (Warde-Farley et al., 2018).

GAIfO: GAIfO learns an adversarial discriminator which encourages distinguishing between agent and prior data, and is then used as an intrinsic reward for the agent policy (Torabi et al., 2018b).

DIAYN: DIAYN learns a skill-conditioned policy by maximizing a diversity based intrinsic reward (Eysenbach et al., 2018b).

PointMass. We consider a simple 2D point mass environment with continuous states and discretized actions to analyse our method on a toy setting. We first gather some prior data that defines what kind of behavior we would want to discover. We consider three kinds of prior data distributions: 1) where the agent starts from the center goes in the four cardinal directions with equal likelihood, 2) where the agent goes to the diagonal directions, and 3) where the agent goes into the four diagonal directions and then bifurcates again into two diverging trajectories . We train an online agent with this prior data and learn to match those behaviors by running Algorithm 1.

Table 1 shows a comparison of the learnt behaviors/skills for all methods across the three different kinds of prior data distributions. We see that methods like DISCERN and GCRL which use goals sampled from the prior data as a way to bootstrap online learning are unable to recover the correct skills, whereas DIAYN recovers skills that are diverse but not aligned with the prior data behavior. GAIfO is able to recover similar behavior as in the prior data, but lacks a notion of individual skills, i.e. the agent in this case learns to traverse any states that are seen in the prior data, while lacking a coherent sense of how two skills differ in their behavior. Finally, our method is able to cluster the prior data into meaningful skills and use them to learn value models (intrinsic reward) to recover similar behavior as in the prior data.

Kitchen from Videos. We use the Kitchen-v3 setup in (Nair et al., 2022) and collect video data by running SAC on the proprioceptive state with an extrinsic reward corresponding to doing multiple tasks such as opening door, turning on light, opening microwave etc. Given this video data, we use a fixed, pretrained R3M representation and use it as a base-level representation. Note that we can learn a representation end-to-end using our objective, however since our focus is on understanding the skill discovery problem, we choose to use a pretrained representation. To that end, we add additional layers to the fixed R3M representation to describe the behavior value model and use them as the

intrinsic reward for a skill conditioned policy, just as in the pointmass experiments. Evaluation trajectories from our approach are presented in Figure 2.

5. Conclusion

We present an approach for recovering and fine-tuning skills from prior video data by learning structured representations of the world. We show our approach is able to better match behaviors and extract interpretable, discrete skills in several PointMass settings relative to past skill discovery and imitation from observation baselines. We additionally test our approach in a simulated kitchen robot manipulation setting. We show our approach is able to extract distinct skills by using video data with further online fine-tuning.

Limitations and Future Work

For the pixel-based Kitchen experiments, we do not recover a one-to-one correspondence between the discrete skill variable and the observed behavior, thus leaving room for improvement towards cleanly segregating the skills and corresponding policy trajectories. Moreover, although our method can learn to discover skills with guidance from video, we did not discuss the setting where the video data has a distribution shift from the online agent. This could involve having cross-embodied agents as well as shifts in observation spaces. Advances in vision-based representation learning could help mitigate this issue. Finally, future work could include video data with more diverse ways to perform certain skills even without the ability to discover some skills, potentially because of a limited action space for the online agent.

References

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight Experience Replay. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018a.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is All You Need: Learning Skills without a Reward Function, October 2018b.

Dibya Ghosh, Chethan Bhateja, and Sergey Levine. Reinforcement learning from passive data via latent intentions. *arXiv preprint arXiv:2304.04782*, 2023.

Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, and Karol Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *arXiv preprint arXiv:1910.11956*, 2019.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Cic: Contrastive intrinsic control for unsupervised skill discovery. *arXiv preprint arXiv:2202.00161*, 2022.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions On Graphics (TOG)*, 41(4):1–17, 2022.

Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. *arXiv preprint arXiv:1907.01657*, 2019.

Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. *arXiv preprint arXiv:1805.01954*, 2018a.

Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral Cloning from Observation, May 2018b.

Faraz Torabi, Sean Geiger, Garrett Warnell, and Peter Stone. Sample-efficient Adversarial Imitation Learning from Observation, June 2019a.

Faraz Torabi, Garrett Warnell, and Peter Stone. Imitation learning from video by leveraging proprioception. *arXiv preprint arXiv:1905.09335*, 2019b.

Faraz Torabi, Garrett Warnell, and Peter Stone. Adversarial imitation learning from state-only demonstrations.

In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2229–2231, 2019c.

Faraz Torabi, Garrett Warnell, and Peter Stone. Recent Advances in Imitation Learning from Observation, June 2019d.

David Warde-Farley, Tom Van de Wiele, Tejas Kulkarni, Catalin Ionescu, Steven Hansen, and Volodymyr Mnih. Unsupervised Control Through Non-Parametric Discriminative Rewards, November 2018.