Visual Adversarial Imitation Learning using Variational Models

Anonymous Authors[1]

Abstract

Reward function specification, which requires considerable human effort and iteration, remains a major impediment for learning behaviors through deep reinforcement learning. In contrast, providing visual demonstrations of desired behaviors presents an easier and more natural way to teach agents. We consider a setting where an agent is provided a fixed dataset of visual demonstrations illustrating how to perform a task, and must learn to solve the task using the provided demonstrations and unsupervised environment interactions. This setting presents a number of challenges including representation learning for visual observations, sample complexity due to high dimensional spaces, and learning instability due to the lack of a fixed reward or learning signal. Towards addressing these challenges, we develop a variational model-based adversarial imitation learning (V-MAIL) algorithm. The model-based approach provides a strong signal for representation learning, enables sample efficiency, and improves the stability of adversarial training by enabling on-policy learning. Through experiments involving several vision-based locomotion and manipulation tasks, we find that V-MAIL learns successful visuomotor policies in a sample-efficient manner, has better stability compared to prior work, and also achieves higher asymptotic performance. We further find that by transferring the learned models, V-MAIL can learn new tasks from visual demonstrations without any additional environment interactions. All results including videos can be found online at `https://sites.google.com/view/variational-mail`.

1. Introduction

The ability of reinforcement learning (RL) agents to autonomously learn by interacting with the environment presents a promising approach for learning diverse skills. However, reward specification has remained a major challenge in the deployment of RL in practical settings (Amodei et al., 2016; Everitt & Hutter, 2019; Rajeswaran et al., 2018). The ability to imitate humans or other expert trajectories allows us to avoid the reward specification problem, while also circumventing challenges related to exploration in RL. Visual demonstrations are also a more natural way to teach robots various tasks and skills in real-world applications. However, this setting is also fraught with a number of technical challenges including representation learning for visual observations, sample complexity due to the high dimensional observation spaces, and learning instability (Portelas et al., 2020; Khetarpal et al., 2020; Lowe et al., 2017) due to lack of a stationary learning signal. We aim to overcome these challenges and to develop an algorithm that can learn from limited demonstration data as well as scale to high-dimensional observation and action spaces often encountered in robotics applications.

Behaviour cloning (BC) is a classic algorithm to imitate expert demonstrations (Pomerleau, 1988), which uses supervised learning to greedily match the expert behaviour at demonstrated expert states. Due to environment stochasticity, covariate shift, and policy approximation error, the agent may drift away from the expert state distribution and ultimately fail to mimic the demonstrator (Ross et al., 2011). While a wide initial state distribution (Spencer et al., 2021) or the ability to interactively query the expert policy (Ross et al., 2011) can circumvent these difficulties, such conditions require additional supervision and are difficult to meet in practical applications. An alternate line of work based on inverse RL (Finn et al., 2016b; Fu et al., 2018) and adversarial imitation learning (Ho & Ermon, 2016; Finn et al., 2016a) aims to not only match actions at demonstrated states, but also the long term visitation distribution (Ghasemipour et al., 2019). These approaches explicitly train a GAN-based classifier (Goodfellow et al., 2014) to distinguish the visitation distribution of the agent from the expert, and use it as a reward signal for training the agent with RL. While these methods have achieved substantial improvement over behaviour cloning without additional expert

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

supervision, they are difficult to deploy in realistic scenarios, primarily due to three reasons: (1) the objective requires on-policy data collection leading to high sample complexity; (2) the non-stationarity reward function changes as the RL agent learns; and (3) high-dimensional observation spaces require representation learning and exacerbate the optimization challenges.

Our main contribution in this work is the development of a new algorithm, variational model-based adversarial imitation learning (V-MAIL), which aims to overcome each of the aforementioned challenges within a single framework. As illustrated in Figure 1, V-MAIL trains a variational latent-space dynamics model and a discriminator that provides a learning reward signal by distinguishing latent rollouts of the agent from the expert. The key insight of our approach is that variational models can address these challenges simultaneously by (a) making it possible to collect on-policy roll-outs inside the model without environment interaction, leading to an efficient and stable optimization process and (b) providing a rich auxiliary objective for efficiently learning compact state representations and which regularizes the discriminator. Furthermore, the variational model also allows V-MAIL to perform zero-shot transfer to new imitation learning tasks. By generating on-policy rollouts within the model, and training the discriminator using these rollouts along with demonstrations of a new task, V-MAIL can learn policies for new tasks without any additional environment interactions.

Through experiments on a collection of vision-based locomotion and manipulation tasks, we find that V-MAIL can learn successful visuomotor control policies through imitation learning. In particular, V-MAIL exhibits stable and near-monotonic learning, is highly sample efficient, and asymptotically matches the expert level performance on most tasks. In contrast, prior algorithms exhibit unstable learning and poor asymptotic performance, often achieving less that 20% of expert level performance. We further show the ability to transfer our models to novel task and acquire qualitatively new behaviors using only a few demonstrations and no additional environment interactions. To our knowledge this is the first approach to use variational model-based training for zero-shot or few-shot imitation learning.

2. Related Work

Here, we review the relevant literature on imitation learning and image-based RL.

Imitation Learning. Recent model-free imitation learning can be categorized as either adversarial or non-adversarial. Adversarial methods inspired by GANs (Goodfellow et al., 2014) train an explicit classifier between expert and policy behaviour and optimize the agent in a two-player minimax game. GAIL (Ho & Ermon, 2016) and AIRL (Fu et al., 2018) are two such algorithms; however they often have poor sample efficiency due to the requirement of on-policy rollouts in the environment. To address sample efficiency issues, off-policy variants such as DAC (Kostrikov et al., 2019) and SAM (Blondé & Kalousis, 2019) have been developed, however they suffer from an objective mismatch when using off-policy data (Kostrikov et al., 2020a), often resulting in learning instability (Blondé et al., 2020).

An alternate line of research attempts to forego adversarial training: SQIL (Reddy et al., 2020) frames the problem as regularized behaviour cloning and trains an off-policy algorithm with rewards of 1 for expert trajectories and 0 for policy ones. RCE (Eysenbach et al., 2021) uses a very similar approach, but derives it as maximizing probability of task success, which they show is equivalent to minimizing the Hellinger distance between the policy occupation distribution and a particular target distribution. ValueDICE (Kostrikov et al., 2020a) uses the same key result for iterative distribution matching as RCE in conjunction with the Donsker-Varadhan representation to obtain an off-policy distribution matching algorithm. In Swamy et al. (2021) the authors derive distribution matching as a bound on policy under-performance, similar to our analysis in Section 4.1 and propose a practical non-adversarial algorithm AdVIL, however in reported experiments it does not outperform behaviour cloning. A few papers have considered model-based imitation learning as well: Baram et al. (2016) is an adversarial algorithm conceptually similar to our approach, but only focuses on low-dimensional state-based tasks and train the discriminator using off-policy replay buffer, which does not allow it to generalize to new tasks. Related to our method is Finn et al. (2016b) which uses a similar reward learning in combination with a locally linear dynamics model, which leads to trajectory centric algorithms and the inability to transfer the model to new tasks. Das et al. (2020) considers a similar setting for inverse RL using a simplified parameterization of the cost function. In this work we develop end-to-end model for adversarial imitation learning in high-dimensional POMDPs and generalization to novel tasks without hand-designed features.

Reinforcement Learning From Images with Variational Models. Reinforcement learning from images is an inherently difficult task, since the agent needs to learn meaningful visual representations to support policy learning. A recent line of research (Gelada et al., 2019; Hafner et al., 2019; Lee et al., 2020; Hafner et al., 2020; Rafailov et al., 2020) train a variational model of the image-based environment as an auxiliary task, either for representation learning only (Gelada et al., 2019; Lee et al., 2020) or for additionally generating on-policy data by rolling out the model (Hafner et al., 2020). Our method builds upon these ideas, but unlike these

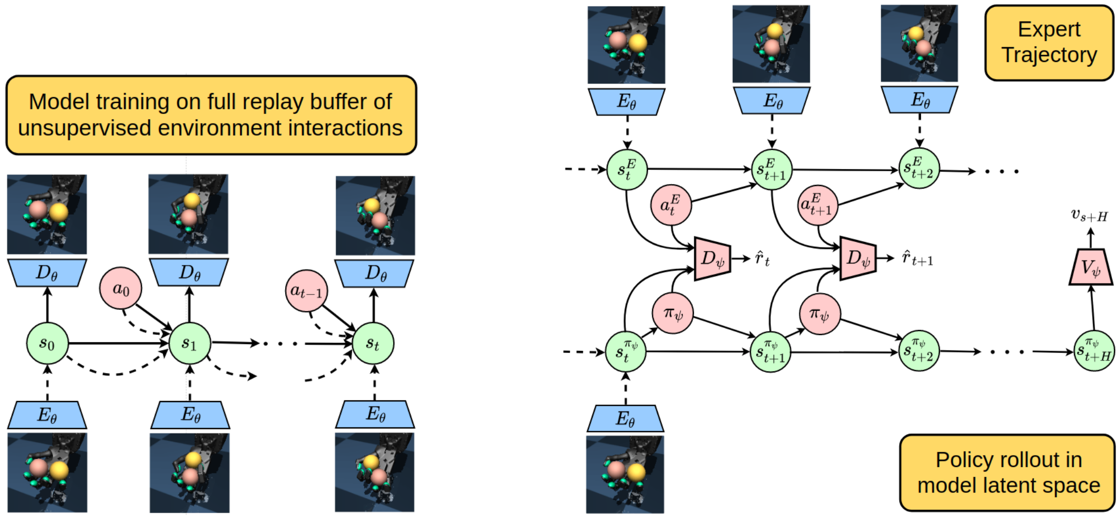

Figure 1. **Left**: the variational dynamics model, which enables joint representation learning from visual inputs and a latent space dynamics model, and the discriminator which is trained to distinguish latent states of expert demonstrations from that of policy rollouts. Dashed lines represent inference and solid lines represent the generative model. **Right**: the policy training, which uses the discriminator as the reward function, so that the policy induces a latent state visitation distribution that is indistinguishable from that of the expert. The learned policy network is composed with the image encoder from the variational model to recover a visuomotor policy.

prior works, considers the problem of learning from visual demonstrations without access to rewards.

3. Preliminaries

We consider the problem setting of learning in partially observed Markov decision processes (POMDPs), which can be described with the tuple: $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{X}, \mathcal{R}, \mathcal{T}, \mathcal{U}, \gamma)$, where $\boldsymbol{s} \in \mathcal{S}$ is the state space, $\boldsymbol{a} \in \mathcal{A}$ is the action space, $\boldsymbol{x} \in \mathcal{X}$ is the observation space and $r = \mathcal{R}(\boldsymbol{s}, \boldsymbol{a})$ is a reward function. The state evolution is Markovian and governed by the dynamics as $\boldsymbol{s}' \sim \mathcal{T}(\cdot|\boldsymbol{s}, \boldsymbol{a})$. Finally, the observations are generated through the observation model $\boldsymbol{x} \sim \mathcal{U}(\cdot|\boldsymbol{s})$. The widely studied Markov decision process (MDP) is a special case of this 7-tuple where the underlying state is directly observed in the observation model.

In this work, we study imitation learning in unknown POMDPs. Thus, we do not have access to the underlying dynamics, the true state representation of the POMDP, or the reward function. In place of the rewards, the agent is provided with a fixed set of expert demonstrations collected by executing an expert policy π^E, which we assume is optimal under the unknown reward function. The agent can interact with the environment and must learn a policy $\pi(\boldsymbol{a}_t|\boldsymbol{x}_{\leq t})$ that mimics the expert.

3.1. Imitation learning as divergence minimization

In line with prior work, we interpret imitation learning as a divergence minimization problem (Ho & Ermon, 2016; Ghasemipour et al., 2019; Ke et al., 2019). For simplicity of exposition, we consider the MDP case in this section, and discuss POMDP extensions in Section 4.2. Let $\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a}) = (1-\gamma)\sum_{t=0}^\infty \gamma^t P(\boldsymbol{s}_t = \boldsymbol{s}, \boldsymbol{a}_t = \boldsymbol{a})$ be the discounted state-action visitation distribution of a policy π in MDP $\mathcal{M}$. Then, a divergence minimization objective for imitation learning corresponds to

$$\min_\pi \ \mathbb{D}(\rho^\pi_\mathcal{M}, \rho^E_\mathcal{M}), \tag{1}$$

where $\rho^E_\mathcal{M}$ is the discounted visitation distribution of the expert policy π^E, and $\mathbb{D}$ is a divergence measure between probability distributions such as KL-divergence, Jensen-Shannon divergence, or a generic $f-$divergence. To see why this is a reasonable objective, let $J(\pi, \mathcal{M})$ denote the expected value of a policy π in $\mathcal{M}$. Inverse RL (Ziebart et al., 2008; Ho & Ermon, 2016; Finn et al., 2016a) interprets the expert as the optimal policy under some unknown reward function. With respect to this unknown reward function, the sub-optimality of any policy π can be bounded as:

$$\left| J(\pi^E, \mathcal{M}) - J(\pi, \mathcal{M}) \right| \leq \frac{R_{\max}}{1-\gamma} \, \mathbb{D}_{TV}(\rho^\pi_\mathcal{M}, \rho^E_\mathcal{M}),$$

since the policy performance is $J(\pi, \mathcal{M}) = \mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim\rho^\pi_\mathcal{M}}\left[r(\boldsymbol{s}, \boldsymbol{a})\right]$. We use $\mathbb{D}_{TV}$ to denote total variation distance. Since various divergence measures are related to the total variation distance, optimizing the divergence between visitation distributions in state space amounts to optimizing a bound on the policy sub-optimality.

3.2. Generative Adversarial Imitation Learning (GAIL)

With the divergence minimization viewpoint, any standard generative modeling technique including density estimation, VAEs, GANs etc. can in principle be used to minimize Eq. 1. However, in practice, use of certain generative modeling techniques can be difficult. A standard density estimation technique would involve directly parameterizing $\rho^\pi_{\mathcal{M}}$, say through auto-regressive flows, and learning the density model. However, a policy that induces the learned visitation distribution in $\mathcal{M}$ is not guaranteed to exist and may prove hard to recover. Similar challenges prevent the direct application of a VAE based generative model as well. In contrast, GANs allow for a policy based parameterization, since it only requires the ability to sample from the generative model and does not require the likelihood. This approach was followed in GAIL, leading to the optimization:

$$\max_{\pi} \min_{D_\psi} \mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim\rho^E_{\mathcal{M}}} \left[-\log D_\psi(\boldsymbol{s},\boldsymbol{a})\right] + \tag{2}$$

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim\rho^\pi_{\mathcal{M}}} \left[-\log\left(1 - D_\psi(\boldsymbol{s},\boldsymbol{a})\right)\right], \tag{3}$$

where D_ψ is a discriminative classifier used to distinguish between samples from the expert distribution and the policy generated distribution. Results from Goodfellow et al. (2014) and Ho & Ermon (2016) suggest that the learning objective in Eq. 2 corresponds to the divergence minimization objective in Eq. 1 with Jensen-Shannon divergence. In order to estimate the second expectation in Eq. 2 we require on-policy samples from π, which is data-inefficient. Adversarial off-policy algorithms, such as (Kostrikov et al., 2019; Blondé & Kalousis, 2019) replace the expectation under the policy distribution with expectation under the current replay buffer distribution, which allows for off-policy training, but no longer guarantee that the policy marginal distribution will match the expert.

4. Variational Model-Based Adversarial Imitation Learning

Generative modeling in the context of imitation learning poses unique challenges. Improving the generative distribution (policy in our case) requires samples from $\rho^\pi_{\mathcal{M}}$, which requires rolling out π in the environment. Furthermore, the complex optimization landscape of a saddle point problem requires many iterations of learning, each of which requires on-policy rollouts. This is unlike typical generative modeling applications where generating samples from the generator is cheap and does not require any environment interactions. To overcome this challenge, we present a model-based imitation learning algorithm. For conceptual clarity and ease of exposition, we will first present our conceptual algorithm in the MDP setting in Section 4.1. Subsequently, we will extend this algorithm to the POMDP case in Section 4.2. Finally, we present a practical version of our algorithm in Section 4.3.

4.1. Model-Based Adversarial Imitation Learning

Model-based algorithms for RL and IL involve learning an approximate dynamics model $\widehat{\mathcal{T}}$ using environment interactions. The learned dynamics model can be used to construct an approximate MDP $\widehat{\mathcal{M}}$. In our context of imitation learning, learning a dynamics model allows us to generate samples from $\widehat{\mathcal{M}}$ as a surrogate for samples from $\mathcal{M}$, leading to the objective:

$$\min_{\pi} \ \mathbb{D}(\rho^\pi_{\widehat{\mathcal{M}}}, \rho^E_{\mathcal{M}}), \tag{4}$$

which can serve as a good proxy to Eq. 1 as long as the model approximation is accurate. In particular, with an α-approximate dynamics model given by $\mathbb{D}_{TV}(\widehat{\mathcal{T}}(\boldsymbol{s},\boldsymbol{a}), \mathcal{T}(\boldsymbol{s},\boldsymbol{a})) \leq \alpha \ \forall(\boldsymbol{s},\boldsymbol{a})$, we can bound the policy suboptimality with respect to the expert as:

$$\left| J(\pi^E, \mathcal{M}) - J(\pi, \mathcal{M}) \right| \tag{5}$$

$$\leq \frac{R_{\max}}{1-\gamma} \, \mathbb{D}_{TV}(\rho^\pi_{\widehat{\mathcal{M}}}, \rho^E_{\mathcal{M}}) + \frac{\alpha \cdot R_{\max}}{(1-\gamma)^2}. \tag{6}$$

Thus, the divergence minimization in Eq. 4 serves as an approximate bound on the sub-optimality with a bias that is proportional to the model error. Thus, we ultimately propose to solve the following saddle point optimization problem:

$$\max_{\pi} \ \min_{D_\psi} \mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim\rho^E_{\mathcal{M}}} \left[-\log D_\psi(\boldsymbol{s},\boldsymbol{a})\right] + \tag{7}$$

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim\rho^\pi_{\widehat{\mathcal{M}}}} \left[-\log\left(1 - D_\psi(\boldsymbol{s},\boldsymbol{a})\right)\right], \tag{8}$$

which requires generating on-policy samples only from the learned model $\widehat{\mathcal{M}}$. We can interleave policy learning according to Eq. 7 with performing policy rollouts in the real environment to iteratively improve the model. Provided the policy is updated sufficiently slowly, Rajeswaran et al. (2020) show that such interleaved policy and model learning corresponds to a stable and convergent algorithm, while being highly sample efficient.

4.2. Extension to POMDPs

In POMDPs, the underlying state is not directly observed, and thus cannot be directly used by the policy. In this case, we typically use the notion of *belief state*, which is defined to be the filtering distribution $P(\boldsymbol{s}_t|\boldsymbol{h}_t)$, where we denote history with $\boldsymbol{h}_t := (\boldsymbol{x}_{\leq t}, \boldsymbol{a}_{<t})$. By using the historical information, the belief state provides more information about the current state, and can enable the learning of better policies. However, learning and maintaining an explicit distribution over states can be difficult. Thus, we consider learning a latent representation of the history $\boldsymbol{z}_t = q(\boldsymbol{h}_t)$, so that $P(\boldsymbol{s}_t|\boldsymbol{h}_t) \approx P(\boldsymbol{s}_t|\boldsymbol{z}_t)$. To develop an algorithm for the POMDP setting, we first make the key observation that imitation learning in POMDPs can be reduced to divergence

Algorithm 1 V-MAIL: Variational Model-Based Adversarial Imitation Learning

```
Require: Expert demos $\mathcal{B}_E$, environment buffer $\mathcal{B}_\pi$.
Randomly initialize variational model $\{q_\theta, \widehat{\mathcal{T}}_\theta\}$, policy $\pi_\psi$ and discriminator $D_\psi$
for number of iterations do
    // Environment Data Collection
    for timestep $t = 1 : T$ do
        Estimate latent state from the belief distribution $\boldsymbol{z}_t \sim q_\theta(\cdot|\boldsymbol{x}_t, \boldsymbol{z}_{t-1}, \boldsymbol{a}_{t-1})$
        Sample action $\boldsymbol{a}_t \sim \pi_\psi(\boldsymbol{a}_t|\boldsymbol{z}_t)$
        Step environment and get observation $\boldsymbol{x}_{t+1}$
    Add data $\{\boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T-1}\}$ to policy replay buffer $\mathcal{B}_\pi$
    for number of training iterations do
        // Dynamics Learning
        Sample a batch of trajectories $\{\boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T-1}\}$ from the joint buffer $\mathcal{B}_E \cup \mathcal{B}_\pi$
        Optimize the variational model $\{q_\theta, \widehat{\mathcal{T}}_\theta\}$ using Equation 11
        // Adversarial Policy Learning
        Sample trajectories from expert buffer $\{\boldsymbol{x}^E_{1:T}, \boldsymbol{a}^E_{1:T-1}\} \sim \mathcal{B}_E$
        Infer expert latent states $\boldsymbol{z}^E_{1:T} \sim q_\theta(\cdot|\boldsymbol{x}^E_{1:T}, \boldsymbol{a}^E_{1:T-1})$ using the belief model $q_\theta$
        Generate latent rollouts $\boldsymbol{z}^{\pi_\psi}_{1:H}$ using the policy $\pi_\psi$ from the forward model $\widehat{\mathcal{T}}_\theta$
        Update the discriminator $D_\psi$ with data $\boldsymbol{z}^E_{1:T}, \boldsymbol{z}^{\pi_\psi}_{1:H}$ using Equation 9
        Update the policy $\pi_\psi$ to improve the value function in Equation 13
```

minimization in the latent belief state representation. To formalize this, we introduce the following theorem. A formal version of the theorem and proof are provided in the appendix.

Theorem 1. *(Divergence bound in latent space; Informal) Consider a POMDP $\mathcal{M}$, and let $\boldsymbol{z}_t$ be a latent space representation of the history and belief state such that $P(\boldsymbol{s}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{a}_{<t}) = P(\boldsymbol{s}_t|\boldsymbol{z}_t)$. Let D_f be a generic $f-$divergence. Then the following inequalities hold:*

$$\begin{aligned} D_f(\rho^\pi_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{a})||\rho^E_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{a})) &\leq D_f(\rho^\pi_{\mathcal{M}}(\boldsymbol{s}, \boldsymbol{a})||\rho^E_{\mathcal{M}}(\boldsymbol{s}, \boldsymbol{a})) \\ &\leq D_f(\rho^\pi_{\mathcal{M}}(\boldsymbol{z}, \boldsymbol{a})||\rho^E_{\mathcal{M}}(\boldsymbol{z}, \boldsymbol{a})) \end{aligned}$$

Theorem 1 suggests that the divergence of visitation distributions in the latent space represents an upper bound of the divergence in the state and observation spaces. This is particularly useful, since we do not have access to the ground-truth states of the POMDP and matching the expert marginal distribution in the high-dimensional observation space (such as images) could be difficult. Furthermore, based on the results in Section 3.1, minimizing the state divergence results in minimizing a bound on policy suboptimality as well. These results provide a direct way to extend the results from Section 4.1 to the POMDP setting. If we can learn an encoder $\boldsymbol{z}_t = q(\boldsymbol{x}_{\leq t}, \boldsymbol{a}_{<t})$ that captures sufficient statistics of the history, and a latent state space dynamics model $\boldsymbol{z}_{t+1} \sim \widehat{\mathcal{T}}(\cdot|\boldsymbol{z}_t, \boldsymbol{a}_t)$, then we can learn the policy by extending Eq. 7 to the induced MDP in the latent space as:

$$\max_{\pi} \min_{D_\psi} \mathbb{E}_{(\boldsymbol{z},\boldsymbol{a})\sim\rho^E_{\mathcal{M}}(\boldsymbol{z},\boldsymbol{a})} \left[-\log D_\psi(\boldsymbol{z}, \boldsymbol{a})\right] + \tag{9}$$

$$\mathbb{E}_{(\boldsymbol{z},\boldsymbol{a})\sim\rho^\pi_{\widehat{\mathcal{M}}}(\boldsymbol{z},\boldsymbol{a})} \left[-\log\left(1 - D_\psi(\boldsymbol{z}, \boldsymbol{a})\right)\right]. \tag{10}$$

Once learned, the policy can be composed with the encoder for deployment in the POMDP.

4.3. Practical Algorithm with Variational Models

The divergence bound of Theorem 1 allows us to develop a practical algorithm if we can learn a good belief state representation. Towards that end we turn to the theory of deep Bayesian filters (Karl et al., 2017) and begin with the likelihood:

$$\log P(\boldsymbol{x}_{1:T}|\boldsymbol{a}_{1:T}) = \log \int \prod_{t=1}^{T} \mathcal{U}(\boldsymbol{x}_t|\boldsymbol{s}_t)\mathcal{T}(\boldsymbol{s}_t|\boldsymbol{a}_{t-1}, \boldsymbol{s}_{t-1})d\boldsymbol{s}_{1:T}$$

We can introduce the belief distribution $q(\boldsymbol{z}_{1:T}|\boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T-1}) = \prod_{t=1}^{T} q(\boldsymbol{z}_t|\boldsymbol{x}_t, \boldsymbol{z}_{t-1}, \boldsymbol{a}_{t-1})$, which considers only model classes that satisfy the the sufficient statistics requirement. Using the introduced belief distribution as the variational distribution, we derive the evidence lower bound (ELBO) (Blei et al., 2016; Kingma & Welling, 2014):

$$\log P(\boldsymbol{x}_{1:T}|\boldsymbol{a}_{1:T}) \geq \mathbb{E}_{q(\boldsymbol{z}_{1:T}|\boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T-1})}\left[\log \prod_{t=1}^{T} \mathcal{U}(\boldsymbol{x}_t|\boldsymbol{z}_t) \frac{\mathcal{T}(\boldsymbol{z}_t|\boldsymbol{a}_{t-1}, \boldsymbol{z}_{t-1})}{q(\boldsymbol{z}_t|\boldsymbol{x}_t, \boldsymbol{z}_{t-1}, \boldsymbol{a}_{t-1})}\right]$$

To estimate the expectation, we can use sequential sampling from the belief distribution $z_t \sim q(\cdot|x_t, z_{t-1}, a_{t-1}), t = 1 : T$ and the reparameterization trick (Kingma & Welling, 2014). This ultimately leads to the empirical variational model training objective:

$$\max_{\theta} \widehat{\mathbb{E}}_{q_\theta}\Big[\sum_{t=1}^{T} \underbrace{\log \widehat{\mathcal{U}}_\theta(x_t|z_t)}_{\text{reconstruction}} - \tag{11}$$

$$\underbrace{\mathbb{D}_{KL}(q_\theta(z_t|x_t, z_{t-1}, a_{t-1})||\widehat{\mathcal{T}}_\theta(z_t|z_{t-1}, a_{t-1}))}_{\text{forward model}}\Big]. \tag{12}$$

That is, we jointly train a belief representation q_θ and a Markovian dynamics model $\widehat{\mathcal{T}}$, which allows us to optimize Eq. 7 in our learned belief space. A number of recent works have considered similar models (Watter et al., 2015; Zhang et al., 2019; Lee et al., 2020; Gelada et al., 2019; Hafner et al., 2019; 2020). We base our architectural choice on the recurrent state space model (Hafner et al., 2019; 2020), as it has shown strong performance in RL tasks from images. In principle, any on-policy RL algorithm can be used to train the policy using Eq. 9. In our setup, the RL objective is a differentiable function of the policy, model, and discriminator parameters. Based on this, we setup a K step value expansion objective (Feinberg et al., 2018; Buckman et al., 2019) given below, and use it for policy learning.

$$V^K_{\theta,\psi}(z_t) = \mathbb{E}_{\pi_\psi, \widehat{\mathcal{T}}_\theta}\Bigg[\sum_{\tau=t}^{t+K-1} \gamma^{\tau-t} \log D_\psi(z^{\pi_\psi}_\tau, a^{\pi_\psi}_\tau) + \gamma^K V_\psi(z^{\pi_\psi}_{t+K})\Bigg] \tag{13}$$

Finally, we train the discriminator D_ψ using Eq. 7 with on-policy rollots from the model $\widehat{\mathcal{T}}$. Our full approach is outlined in Algorithm 1.

4.4. Zero-Shot Transfer to New Imitation Tasks

Our model-based approach is well suited to the problem of zero-shot transfer to new imitation learning tasks, i.e. transferring to a new task using a modest number of demonstrations and no additional samples collected in the environment.. In particular, we assume a set of source tasks $\{\mathcal{T}^i\}$, each with a buffer of expert demonstrations $\mathcal{B}^i_E$. Each source task corresponds to a different POMDP with different underlying rewards, but shared dynamics. The underlying state space may also change across tasks, but the dynamics and observation model are shared across tasks. During training, the agent can interact with each source environment and collect additional data. At test time, we're introduced with a new target task $\mathcal{T}$ with corresponding expert demonstrations $\mathcal{B}_E$ and the goal is to obtain a policy that achieves high reward without additional interaction with the environment.

Algorithm 2 Zero-Shot Transfer with V-MAIL

```
Require: Expert demos $\mathcal{B}^i_E$ for each source task, expert demos $\mathcal{B}_E$ for target task
Randomly initialize policy $\pi_\psi$, and discriminator $D_\psi$
Train Alg 1 on source tasks, yielding shared model $\{q_\theta, \widehat{\mathcal{T}}_\theta\}$ and aggregated replay buffer $\mathcal{B}_\pi$
for number of training iterations do
    // Dynamics Fine-Tuning using Expert Trajectories
    Update the variational model $\{q_\theta, \widehat{\mathcal{T}}_\theta\}$ using Equation 11 with data from $\mathcal{B}_E \cup \mathcal{B}_\pi$
    // Adversarial Policy Learning
    Update discriminator $D_\psi$ and policy $\pi_\psi$ with Equations 9 and 13.
```

Our key observation is that we can optimize Eq. 9 under our model and still obtain an upper bound on policy sub-optimality via Eq. 5. Furthermore, the sub-optimality is bound by the accuracy of our model over the marginal state-action distribution of the target task expert. Specifically, we first train on all of the source tasks using Algorithm 1, training a single shared variational model across the tasks. By fine-tuning that model on data that includes the target task expert demonstrations our hope is that we can get an accurate model and thus a high-quality policy. Similarly to Algorithm 1, we then train a discriminator and policy for the target task using only model rollouts. This approach is outlined in Algorithm 2.

5. Experiments

In our experiments, we aim to answer several questions: (1) can V-MAIL successfully scale to environments with image observations, (2) how does V-MAIL compare to state of the art model-free imitation approaches, (3) can V-MAIL solve realistic manipulation tasks and environments with complex physical interactions, and (4) can V-MAIL enable zero-shot transfer to new tasks? All experiments were carried out on a single Titan RTX GPU using an internal cluster for about 1000 GPU hours.

5.1. Single-Task Experiments

Comparisons. To answer question (2), we choose to compare V-MAIL to model-free adversarial and non-adversarial imitation learning methods. For the former, we choose DAC (Kostrikov et al., 2019) as a representative approach, which we equip with DrQ data augmentation for greater performance on vision-based tasks. For the latter, we consider SQIL (Reddy et al., 2020), also equipped with DrQ training. We refer to each approach with data augmentation as DA-DAC and DA-SQIL respectively. Both of these methods are off-policy algorithms, which we expect to be

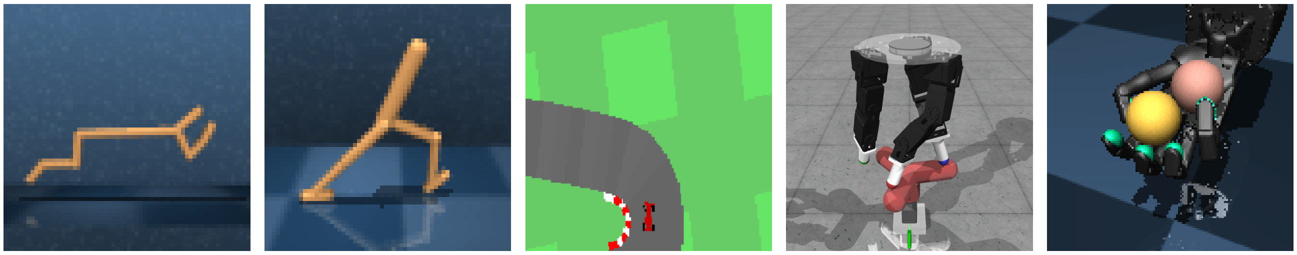

Figure 2. Illustration of the environments used in our experiments: Cheetah, Walker, Car Racing, D'Claw, and Baoding Balls. In all environments, the agent has access only to the RGB image frames as observations, except with additional access to proprioception in the Baoding Balls environment.

considerably more sample efficient than on-policy methods like GAIL (Ho & Ermon, 2016) and AIRL (Fu et al., 2018).

Environments and Demonstration Data. To answer the above questions, we consider the five visual control environments illustrated in Figure 2. We first evaluate our method on the visual Cheetah and visual Walker tasks from the DeepMind Control Suite (Tassa et al., 2018). Following SQIL (Reddy et al., 2020) we also consider the classic Car Racing environment, which is difficult to solve even with ground-truth rewards. In addition, we benchmark our method on a custom D'Claw environment from the Robel suite (Ahn et al., 2019), entirely from images without proprioception. This makes the task challenging due to a complex action dynamics, contact dynamics, and occlusions from the robot fingers. Our final environment is the Baoding balls task from Nagabandi et al. (2019). This is an extremely challenging task for policy learning, even in the state-based case. All tasks are from raw RGB images, while the Baoding balls task additionally includes robot proprioception. All methods receive access to use 10 expert demonstrations, with the exception of the Baoding environment, which uses 25 demonstrations. The demonstrations for the DeepMind Control and D'Claw tasks are generated using a policy trained with SAC (Haarnoja et al., 2018), the expert data for the Car Racing environment is generated using Dreamer (Hafner et al., 2020), and the demonstrations for the Baoding task is generated using PDDM (Nagabandi et al., 2019) from low-dimensional states. Additional details on the experimental set-up are provided in the appendix.

Results. Experiment results are shown in Figure 3. To answer questions (1) and (2), we compare V-MAIL to DA-SQIL and DA-DAC on the Cheetah and Walker tasks. We find that V-MAIL efficiently and reliably solves both tasks; in contrast, the model-free methods initially outperform V-MAIL, but their performance has high variance across random seeds and exhibits significant instability. Such stability issues have also been observed by Swamy et al. (2021), which provides some theoretical explanation in the case of SQIL and the suggestion of early stopping as a mitigation technique. In the case of DAC, the reasons for instability are less clear. Motivated by instability we observed in the critic loss for DA-DAC, we experimented with a number of mitigation strategies in an attempt to improve DA-DAC, including constraining the discriminator, varying the buffer and batch sizes, and separating the convolutional encoders of the discriminator and the actor/critic; however, these techniques didn't fully prevented the degradation in performance.

On the Car Racing environment, we find that DA-SQIL and DA-DAC can reach or outperform behavior cloning, but struggle to reach expert-level performance. In contrast, V-MAIL stably and reliably achieves near-expert performance in about 200k environment steps. Note that Reddy et al. (2020) report expert-level performance on this task, but in an easier setting with double the number of expert demonstrations available (20 vs. 10). Given that tracks are randomly generated per episode demanding significant generalization, it is not surprising that the problem becomes considerably more difficult with only 10 demonstrations.

Finally, to answer question (3), we consider the D'Claw and Baoding Balls tasks. In the D'Claw environment, SQIL fails to make progress, while DA-DAC makes significant progress initially but quickly degrades. V-MAIL solves the task in less than 100k environment steps. In the most challenging visual Baoding Balls problem, involving a 26-dimensional control space, V-MAIL is the only algorithm to reach any success.

5.2. Transfer Experiments

Transfer Scenarios. To evaluate V-MAIL's ability to learn new imitation tasks in a zero-shot way (i.e. without any additional environment samples) we deploy Algorithm 2 on two domains: in a locomotion experiment we train on the Walker Stand and Walker Run (target speed greater than 8) tasks and and evaluate transfer to the Walker Walk (target speed between 2 and 4) task from the DeepMind Control suite. In a manipulation scenario, we use a set of custom D'Claw Screw tasks from the Robel suite (Ahn et al., 2019). We train our model on the 3-prong tasks with clockwise and counter-clockwise rotation, as well as the 4-prong task

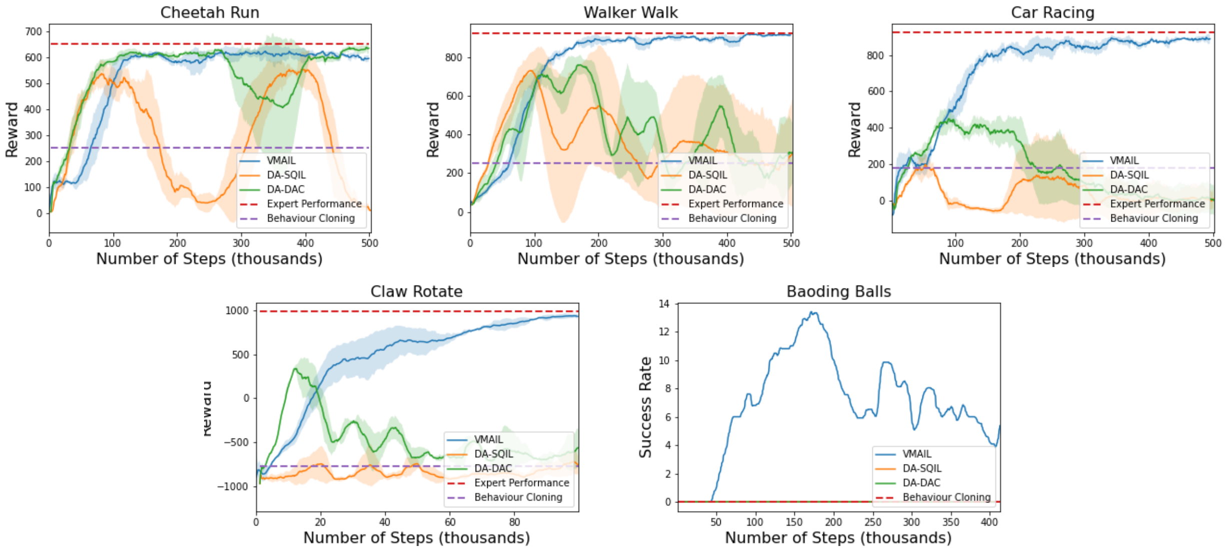

Figure 3. Learning curves showing ground truth reward versus number of environment steps for V-MAIL (ours), prior model-free imitation learning approaches, and behavior cloning on five visual imitation tasks. We find that V-MAIL consistently outperforms prior methods in terms of sample efficiency, final performance, and stability, particularly for the first four environments where V-MAIL reaches near-expert performance. In the most challenging visual Baoding Balls task, which is notably difficult even with ground-truth state, only V-MAIL is able to make some progress, but all methods struggle. Confidence intervals are shown with 1 SD over 3 runs.

with counter-clockwise rotation and evaluate transfer to the 4-prong task with clockwise rotation.

Comparisons. To our knowledge, no prior work has considered this zero-shot transfer scenario previously. Thus, we devise several points of comparison. First, we compare to directly applying the policy learned in the most related source task to the target task. This tests whether the target task demands qualitatively distinct behavior. Second, we compare to an offline version of DAC, augmented with the CQL approach (Kumar et al., 2020), where samples collected from the source task are used to update the policy, with the target task demonstrations used to learn the reward. Finally, we also compare to behavior cloning on the target task demonstrations (without leveraging any source task data), and an oracle that performs V-MAIL on the target task directly.

Method	Walker Walk	Claw Rotate
Offline DAC	8.8%	-0.7%
Behavior cloning	26.8%	8.3%
Policy transfer	21.3%	5.6%
V-MAIL (ours)	**92.7%**	**97.9%**
Target task IL (oracle)	98.2%	102.3%

Table 1. Performance on zero-shot transfer to a new imitation learning task as percent of expert return. Each method is provided with 10 demonstrations of the target task, and zero additional samples in the environment. V-MAIL can solve the target tasks within its learned model without any additional samples, while model-free transfer learning approaches fail.

Results. Our results are shown in Table 1. Policy transfer performs poorly, suggesting that the target task indeed requires qualitatively different behaviour from the few training tasks available. Further, behavior cloning on the target demonstrations is not sufficient to learn the task. Offline DAC also shows poor performance. Finally, we see that V-MAIL almost matches the performance of the agent explicitly trained on task, indicating the learned model and the algorithm for training within that model can be used not just for efficient visual imitation learning, but also for zero-shot transfer to new tasks.

6. Conclusion

In this work we presented V-MAIL, a model-based imitation learning algorithm that works from high-dimensional image observations. V-MAIL learns a model of the environment, which serves a strong supervision signal for visual representation learning, as well as allowing us to train an imitation learning algorithm on-policy, without sacrificing sample efficiency. V-MAIL achieves better asymptotic returns, is more stable, and matches the sample efficiency of off-policy model-free approaches. We also find that by training a policy using only model rollouts, our approach is a strong procedure for zero-shot transfer to novel imitation learning tasks.

References

Ahn, M., Zhu, H., Hartikainen, K., Ponte, H., Gupta, A., Levine, S., and Kumar, V. ROBEL: RObotics BEnchmarks for Learning with low-cost robots. In *Conference on Robot Learning (CoRL)*, 2019.

Ali, S. M. and Silvey, S. . a general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society*, 28(1):131:142, 1966.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P. F., Schulman, J., and Mané, D. Concrete problems in ai safety. *ArXiv*, abs/1606.06565, 2016.

Baram, N., Anschel, O., and Mannor, S. Model-based adversarial imitation learning. *Conference on Neural Information Processing Systems*, 2016.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112:859 – 877, 2016.

Blondé, L. and Kalousis, A. Sample-efficient imitation learning via generative adversarial nets. *AISTATS*, 2019.

Blondé, L., Strasser, P., and Kalousis, A. Lipschitzness is all you need to tame off-policy generative adversarial imitation learning, 2020.

Buckman, J., Hafner, D., Tucker, G., Brevdo, E., and Lee, H. Sample-efficient reinforcement learning with stochastic ensemble value expansion. *Conference on Neural Information Processing Systems*, 2019.

Das, N., Bechtle, S., Davchev, T., Jayaraman, D., Rai, A., and Meier, F. Model-based inverse reinforcement learning from visual demonstrations. *Conference on Robot Learning*, 2020.

Everitt, T. and Hutter, M. Reward tampering problems and solutions in reinforcement learning: A causal influence diagram perspective. *ArXiv*, abs/1908.04734, 2019.

Eysenbach, B., Levine, S., and Salakhutdinov, R. Replacing rewards with examples: Example-based policy search via recursive classification, 2021.

Feinberg, V., Wan, A., Stoica, I., Jordan, M. I., Gonzalez, J. E., and Levine, S. Model-based value estimation for efficient model-free reinforcement learning. *International Conference on Machine Learning*, 2018.

Finn, C., Christiano, P., Abbeel, P., and Levine, S. A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models. *ArXiv Preprint*, 2016a.

Finn, C., Levine, S., and Abbeel, P. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pp. 49–58. PMLR, 2016b.

Fu, J., Luo, K., and Levine, S. Learning robust rewards with adversarial inverse reinforcement learning. *International Conference on Learning Representations*, 2018.

Gelada, C., Kumar, S., Buckman, J., Nachum, O., and Bellemare, M. G. Deepmdp: Learning continuous latent space models for representation learning. *International Conference on Machine Learning*, 2019.

Ghasemipour, S. K. S., Zemel, R., and Gu, S. A divergence minimization perspective on imitation learning methods. *Conference on Robot Learning*, 2019.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In *NIPS*, 2014.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *International Conference on Machine Learning*, 2018.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. *International Conference on Machine Learning*, 2019.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. *International Conference on Learning Representations*, 2020.

Ho, J. and Ermon, S. Generative adversarial imitation learning. *Conference on Neural Information Processing Systems*, 2016.

Karl, M., Soelch, M., Bayer, J., and van der Smagt, P. Deep variational bayes filters: Unsupervised learning of state space models from raw data. *International Conference on Machine Learning*, 2017.

Ke, L., Barnes, M., Sun, W., Lee, G., Choudhury, S., and Srinivasa, S. Imitation learning as f-divergence minimization. *ArXiv*, abs/1905.12888, 2019.

Khetarpal, K., Riemer, M., Rish, I., and Precup, D. Towards continual reinforcement learning: A review and perspectives. *ArXiv*, abs/2012.13490, 2020.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes, 2014.

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

Kostrikov, I., Nachum, O., and Tompson, J. Imitation learning via off-policy distribution matching. *International Conference on Learning Representations*, 2020a.

Kostrikov, I., Yarats, D., and Fergus, R. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels, 2020b.

Kumar, A., Zhou, A., Tucker, G., and Levine, S. Conservative q-learning for offline reinforcement learning. *Conference on Neural Information Processing Systems*, 2020.

Lee, A. X., Nagabandi, A., Abbeel, P., and Levine, S. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. *Conference on Neural Information Processing Systems*, 2020.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. In *NIPS*, 2017.

Nagabandi, A., Konolige, K., Levine, S., and Kumar, V. Deep dynamics models for learning dexterous manipulation. *Conference on Robot Learning*, 2019.

Pomerleau, D. A. Alvinn: an autonomous land vehicle in a neural network. In *Proceedings of the 1st International Conference on Neural Information Processing Systems*, pp. 305–313, 1988.

Portelas, R., Colas, C., Weng, L., Hofmann, K., and Oudeyer, P.-Y. Automatic curriculum learning for deep rl: A short survey. *ArXiv*, abs/2003.04664, 2020.

Rafailov, R., Yu, T., Rajeswaran, A., and Finn, C. Offline reinforcement learning from images with latent space models. *arXiv preprint arXiv:2012.11547*, 2020.

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

Rajeswaran, A., Mordatch, I., and Kumar, V. A Game Theoretic Framework for Model-Based Reinforcement Learning. In *ICML*, 2020.

Reddy, S., Dragan, A. D., and Levine, S. Sqil: Imitation learning via reinforcement learning with sparse rewards. *International Conference on Learning Representations*, 2020.

Ross, S., Gordon, G. J., and Bagnell, J. A. A reduction of imitation learning and structured prediction to no-regret online learning. *AISTATS*, 2011.

Spencer, J., Choudhury, S., Venkatraman, A., Ziebart, B., and Bagnell, J. A. Feedback in imitation learning: The three regimes of covariate shift. *ArXiv Preprint*, 2021.

Swamy, G., Choudhury, S., Wu, Z. S., and Bagnell, J. A. Of moments and matching: Trade-offs and treatments in imitation learning. 2021.

Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., de Las Casas, D., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T., and Riedmiller, M. Deepmind control suite, 2018.

Watter, M., Springenberg, J. T., Boedecker, J., and Riedmiller, M. Embed to control: A locally linear latent dynamics model for control from raw images, 2015.

Zhang, M., Vikram, S., Smith, L., Abbeel, P., Johnson, M. J., and Levine, S. Solar: Deep structured representations for model-based reinforcement learning. *International Conference on Machine Learning*, 2019.

Ziebart, B. D., Maas, A. L., Bagnell, J., and Dey, A. Maximum entropy inverse reinforcement learning. In *AAAI*, 2008.

A. Theoretical Proofs

We base our approach on the main theoretical result of the paper:

Theorem 2. *Consider a POMDP $\mathcal{M}$, and let $\boldsymbol{z}_t$ be a latent space representation of the history and belief state such that $P(\boldsymbol{s}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{a}_{<t}) = P(\boldsymbol{s}_t|\boldsymbol{z}_t)$. Also assume that $P(\boldsymbol{s}_t|\boldsymbol{z}_t, \boldsymbol{a}_t) = P(\boldsymbol{s}_t|\boldsymbol{z}_t)$. Let D_f be a generic $f-$divergence. Then the following inequalities hold:*

$$D_f(\rho^\pi_\mathcal{M}(\boldsymbol{x}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{x}, \boldsymbol{a})) \leq D_f(\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})) \leq D_f(\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a}))$$

Proof. The condition $P(\boldsymbol{s}_t|\boldsymbol{z}_t, \boldsymbol{a}_t) = P(\boldsymbol{s}_t|\boldsymbol{z}_t)$ essentially states that the belief distribution is independent of the policy or that the actions of both the agent and the expert do not carry additional information about the state. This will be true of both agents are trained using the belief, without access to the ground truth state. We should note that a similar assumption would be needed to prove the first inequality namely: $\mathcal{U}(\boldsymbol{x}_t|\boldsymbol{s}_t, \boldsymbol{a}_t) = \mathcal{U}(\boldsymbol{x}_t|\boldsymbol{s}_t)$, however this holds from the structure of the POMDP. With these assumptions, the proof is straightforward application of the following data-processing inequality (Ali & Silvey, 1966). We will prove that $D_f(\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})) \leq D_f(\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a}))$:

$$D_f(\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})) = \mathbb{E}_{\boldsymbol{z},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{z},\boldsymbol{a})}\left[f\left(\frac{\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})}\right)\right] = \tag{14}$$

$$\mathbb{E}_{\boldsymbol{z},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{z},\boldsymbol{a})}\mathbb{E}_{\boldsymbol{s}\sim P(\boldsymbol{s}|\boldsymbol{z})}\left[f\left(\frac{\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{a})}\frac{P(\boldsymbol{s}|\boldsymbol{z})}{P(\boldsymbol{s}|\boldsymbol{z})}\right)\right] = \tag{15}$$

$$\mathbb{E}_{\boldsymbol{z},\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{z},\boldsymbol{s},\boldsymbol{a})}\left[f\left(\frac{\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}\right)\right] = \tag{16}$$

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{s},\boldsymbol{a})}\left[\mathbb{E}_{\boldsymbol{z}\sim\rho^E_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})}f\left(\frac{\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}\right)\right] \geq \tag{17}$$

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{s},\boldsymbol{a})}\left[f\left(\mathbb{E}_{\boldsymbol{z}\sim\rho^E_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})}\frac{\rho^\pi_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{z}, \boldsymbol{s}, \boldsymbol{a})}\right)\right] = \tag{18}$$

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{s},\boldsymbol{a})}\left[f\left(\mathbb{E}_{\boldsymbol{z}\sim\rho^E_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})}\frac{\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})\rho^\pi_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})\rho^E_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a})}\right)\right] = \tag{19}$$

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{s},\boldsymbol{a})}\left[f\left(\mathbb{E}_{\boldsymbol{z}\sim\rho^\pi_\mathcal{M}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})}\frac{\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})}\right)\right] = \tag{20}$$

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\rho^E_\mathcal{M}(\boldsymbol{s},\boldsymbol{a})}\left[f\left(\frac{\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})}{\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})}\right)\right] = \tag{21}$$

$$D_f(\rho^\pi_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})||\rho^E_\mathcal{M}(\boldsymbol{s}, \boldsymbol{a})) \tag{22}$$

The first two equalities (10-11) follow from the fact that $\rho^\pi_\mathcal{M}(\boldsymbol{s}|\boldsymbol{z}, \boldsymbol{a}) = P(\boldsymbol{s}|\boldsymbol{z}) = \rho^E_\mathcal{M}(\boldsymbol{s}|\boldsymbol{z}, \boldsymbol{a})$ from the assumptions of the Theorem. The inequality (12) is a direct application of Jensen's inequality and the definition of an $f-$divergence. The other part of the main result follows the same reasoning, considering the observation model $\mathcal{U}(\boldsymbol{x}|\boldsymbol{s})$, rather than the belief distribution $P(\boldsymbol{s}|\boldsymbol{z})$. □

B. Practical Off-Policy Imitation Learning Algorithms

Training reinforcement learning policies from images is challenging using environment rewards, but even more so in the case of adversarial imitation learning. We explicitly choose to benchmark our method against SQIL and DAC, which use sample efficient off-policy training. In addition we can augment these approaches with state of the art method DrQ (Kostrikov et al., 2020b), which has shown up to two orders of magnitude improvement in sample efficiency when training policies from raw pixels. The key of the DrQ approach is to introduce a family of image-augmentation functions $f(\boldsymbol{s}, v)$, where $\boldsymbol{s}$ is an environment state (a set of stacked images) and v are augmentation parameters, from a fixed set of transformations. Given a batch of transition tuples $(\boldsymbol{s}_i, \boldsymbol{a}_i, \boldsymbol{s}'_i, \boldsymbol{r}_i)$ the standard Q-learning procedure is augmented as follows: the target values for the Bellman backups are computed as:

$$\boldsymbol{y}_i = \boldsymbol{r}_i + \gamma \frac{1}{K} \sum_{k=1}^{K} Q_\theta^{target}(f(\boldsymbol{s}'_i, v'_{i,k}), \boldsymbol{a}'_{i,k}) \text{ where } \boldsymbol{a}'_{i,k} \sim \pi(\cdot | f(\boldsymbol{s}'_i, v'_{i,k})) \tag{23}$$

while the Q-function is updated by:

$$\theta \leftarrow \theta - \lambda \nabla_\theta \frac{1}{NM} \sum_{i=1,m=1}^{N,M} (Q_\theta(f(\boldsymbol{s}_i, v_{i,m}), \boldsymbol{a}_i) - \boldsymbol{y}_i)^2 \tag{24}$$

We can directly adapt SQIL to this setup, by using stationary rewards for the expert and policy replay buffers. For DAC, we train an additional discriminator D_ψ minimizing the objective:

$$\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}^E}\left[\frac{1}{K}\sum_{k=1}^{K} -\log D_\psi(f(\boldsymbol{s}, v_k), \boldsymbol{a})\right] + \mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}^\pi}\left[\frac{1}{K}\sum_{k=1}^{K} -\log(1 - D_\psi(f(\boldsymbol{s}, v_k), \boldsymbol{a}))\right] \tag{25}$$

we then train the DAC with a modified version of Eq. 24:

$$\boldsymbol{y}_i = \frac{1}{K}\sum_{k=1}^{K} \log D_\psi(f(\boldsymbol{s}_i, v_{i,k}), \boldsymbol{a}_i) + \gamma Q_\theta^{target}(f(\boldsymbol{s}'_i, v'_{i,k}), \boldsymbol{a}'_{i,k}) \tag{26}$$

In our implementation the discriminator, critic and policy share the same convolutional encoder, which is trained using the discriminator and critic loss only. During training of this baseline, we noticed that periods of poor performance coincide with instability in the critic loss, rather than the discriminator. We hypothesise that this is caused by issues with value function bootstrapping with non-stationary rewards. We experimented with a number of mitigation strategies in an attempt to improve performance of this baseline, including constraining the discriminator, different regularization techniques, varying the buffer and batch sizes and separating the convolutional encoders of the discriminator and the actor/critic; however, these techniques didn't fully prevented the degradation in performance.