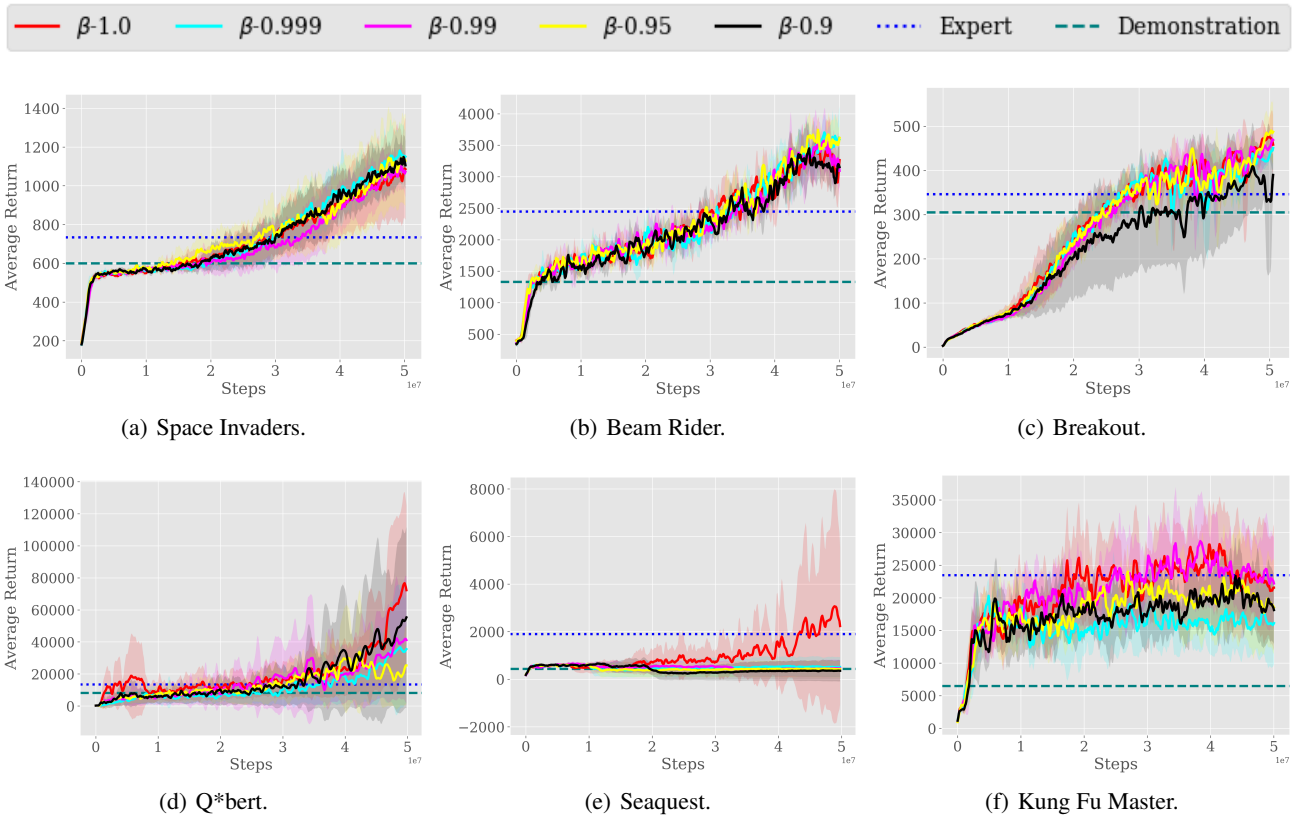
A. Appendix

A.1. Ablation study

A.1.1. Ablation study of our method with different β - learning curves.

The full learning curves of our method with different β have been shown on Figure 7.



(a) Space Invaders. (b) Beam Rider. (c) Breakout.

(d) Q*bert. (e) Seaquest. (f) Kung Fu Master.

Figure 7. Average return vs. number of simulation steps on Atari games. The solid lines show the mean performance over 5 random seeds. The shaded area represents the standard deviation from the mean. The blue dotted line denotes the average return of expert. The area above the blue dotted line means performance beyond the expert.

A.1.2. The effect of standardization in GIRIL and CDIL

Table 6. Ablation study of standardized intrinsic reward on the GIRIL and CDIL. The results shown are the mean performance over 5 random seeds with better-than-expert performance in bold.

	Expert	Demonstration	GIRIL		CDIL	
Game	Average	Average	Standardized	Original	Standardized	Original
Space Invaders	734.1	600.0	**992.9**	565.5	668.9	532.7
Beam Rider	2,447.7	1,332.0	**3,202.3**	1,810.4	**2,556.9**	1,808.1
Breakout	346.4	305.0	**426.9**	**375.2**	**369.2**	**369.7**
Q*bert	13,441.5	8,150.0	**42,705.7**	**21,080.3**	**30,070.8**	12,755.4
Seaquest	1,898.8	440.0	731.8	**2,022.4**	897.7	775.4
Kung Fu Master	23,488.5	6,500.0	**23,543.6**	**23,984.8**	17,291.6	18,663.6

Table 6 compares GIRIL and CDIL trained via PPO with the standardized intrinsic reward and the original intrinsic reward. With the original intrinsic reward, CDIL was able to outperform the one-life demonstration on five out of six games, but only beat the expert on Breakout. With standardization, CDIL was able to surpass the expert in two more games, Beam

Rider and Q*bert. GIRIL maintain its superior performance with better-than-one-life performance on five of six games, and better-than-expert performance on four. Notably, standardizing the reward gave GIRIL the power to outperform the one-life results with two more games and the expert results with one more game. Without standardization, GIRIL still outperformed other baselines.

A.1.3. THE EFFECTS OF r_t IN GAIL AND I_c IN VAIL

We then compare GIRIL against GAIL with two different reward function r_t ($r_t^{(1)}$=$-\log(D(s_t, a_t))$ and $r_t^{(2)}$=$-\log(1 - D(s_t, a_t))$, where D is the *discriminator*) and VAIL with two different information constraints I_c (I_c=0.2, and I_c=0.5). I_c=0.2 and I_c=0.5 are the default hyper-parameters in Karnewar (2018) and Peng et al. (2019), respectively. The results are provided in Table 7.

Table 7. Parameter Analysis of the GIRIL versus VAIL with different information constraints I_c, and versus GAIL with different rewards r_t, i.e. $r_t^{(1)}$=$-\log(D(s_t, a_t))$ and $r_t^{(2)}$=$-\log(1 - D(s_t, a_t))$. The results shown are the mean performance over 5 random seeds with better-than-expert performance in bold.

	Expert	Demonstration	GIRIL	VAIL (I_c)		GAIL (r_t)	
Game	Average	Average	Average	0.2	0.5	$r_t^{(1)}$	$r_t^{(2)}$
Space Invaders	734.1	600.0	**992.9**	549.4	426.5	228.0	129.9
Beam Rider	2,447.7	1,332.0	**3,202.3**	**2,864.1**	**2,502.7**	285.5	131.3
Breakout	346.4	305.0	**426.9**	36.1	27.2	1.3	2.5
Q*bert	13,441.5	8,150.0	**42,705.7**	10,862.3	**54,247.3**	8,737.4	205.3
Seaquest	1,898.8	440.0	**2,022.4**	312.9	1,746.7	0.0	28.9
Kung Fu Master	23,488.5	6,500.0	**23,543.6**	**24,615.9**	14,709.3	1,324.5	549.7

As the results show, GAIL with $-\log(D(s_t, a_t))$ performed better than that with $-\log(1 - D(s_t, a_t))$. VAIL showed similar performance no matter the information constraint. Both outplayed the expert on two games - an overall worse performance than CDIL with standardized reward and GIRIL with both types of reward.

A.1.4. THE EFFECT OF THE NUMBER OF FULL-EPISODE DEMONSTRATIONS.

We also evaluated our method with different number of full-episode demonstrations on both Atari games and continuous control tasks. Table 8 and Table 9 show the detailed quantitative comparison of imitation learning methods across different number of full-episode demonstrations in the games, Breakout and Space Invaders. The comparisons on two continuous control tasks, InvertedPendulum and InvertedDoublePendulum, have been shown in Table 10 and Table 11.

The results shows that our method GIRIL achieves the highest performance across different numbers of full-episode demonstrations, and CDIL usually comes the second best. GAIL is able to achieve better performance with the increase of the demonstration number in both continuous control tasks.

Table 8. Parameter Analysis of the GIRIL versus other baselines with different number of full-episode demonstrations on Breakout game. The results shown are the mean performance over 5 random seeds with best performance in bold.

# Demonstrations	GIRIL	CDIL	VAIL	GAIL
1	**413.9**	361.2	34.0	1.4
5	**384.4**	334.9	30.5	1.9
10	**415.0**	332.1	27.1	2.9

Table 9. Parameter Analysis of the GIRIL versus other baselines with different number of full-episode demonstrations on Space Invaders game. The results shown are the mean performance over 5 random seeds with best performance in bold.

# Demonstrations	GIRIL	CDIL	VAIL	GAIL
1	**1,073.8**	557.5	557.0	190.0
5	**977.6**	580.6	4.4	190.0
10	**910.3**	533.2	90.0	190.0

Table 10. Parameter Analysis of the GIRIL versus other baselines with different number of full-episode demonstrations on InvertedPendulum task. The results shown are the mean performance over 5 random seeds with best performance in bold.

# Demonstrations	GIRIL	CDIL	VAIL	GAIL
1	**990.2**	979.7	113.6	612.6
5	**1,000.0**	1,000.0	78.5	1,000.0
10	994.4	**999.9**	80.1	988.2

Table 11. Parameter Analysis of the GIRIL versus other baselines with different number of full-episode demonstrations on InvertedDoublePendulum task. The results shown are the mean performance over 5 random seeds with best performance in bold.

# Demonstrations	GIRIL	CDIL	VAIL	GAIL
1	**9,164.9**	7,114.7	725.2	1,409.0
5	**9,290.4**	7,628.7	342.9	8,634.5
10	**8,972.8**	8,548.6	714.8	8,842.0

A.2. Details of the *curiosity-driven imitation learning* (CDIL)

The Intrinsic Curiosity Module (ICM) is a natual choice for reward learning in imitaiton learning. ICM is a state-of-the-art exploration method (Pathak et al., 2017; Burda et al., 2019) that transforms high-dimensional states into a visual feature space and then impose a cross-entropy loss and a Euclidean loss to learn the features with a self-supervised inverse dynamics model. Further, the prediction error in the feature space becomes the intrinsic reward function for exploration. As illustrated in Figure 8, ICM encodes the states s_t, s_{t+1} into features and then the inverse dynamics model g_{θ_I} is trained to predict actions from the states features $\phi(s_t)$ and $\phi(s_{t+1})$. Additionally, the forward model f_{θ_F} takes a feature $\phi(s_t)$ and an action a_t as input and predicts the feature representation of state s_{t+1}. The intrinsic reward is calculated as the curiosity, i.e. the prediction error in the feature space.

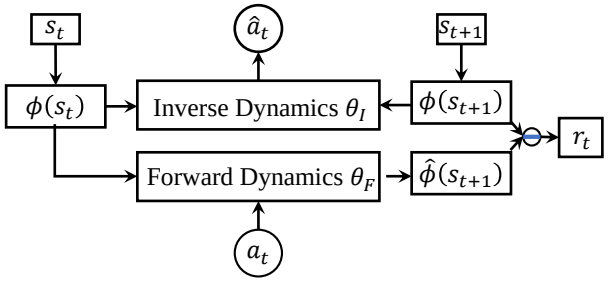


Figure 8. Intrinsic Curiosity Module (ICM).

In ICM, the inverse dynamics model is used to predict the action $\hat{a}_t = g_{\theta_I}(\phi(s_t), \phi(s_{t+1}))$, and is optimized by:

$$\min_{\theta_I} L_I(\hat{a}_t, a_t), \tag{3}$$

where L_I is the loss function measures the discrepancy between the predicted and actual action. In our experiments, we use cross-entropy loss for Atari games and mean squared error (MSE) for continuous control tasks.

The forward dynamics model estimates the feature of next state $\hat{\phi}(s_{t+1}) = f_{\theta_F}(\phi(s_t), a_t)$, and is optimized by:

$$\min_{\theta_F} L_F(\phi(s_t), \hat{\phi}(s_{t+1})) = \|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\|_2^2, \tag{4}$$

where $\|\cdot\|_2$ is the L2 norm.

ICM is optimized by minimizing the overall objective as follows:

$$\min_{\theta_I, \theta_F} L_I + L_F \tag{5}$$

The intrinsic reward signal r_t is calculated as the prediction error in feature space:

$$r_t = \lambda \|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\|_2^2 \tag{6}$$

where $\|\cdot\|_2$ is the L2 norm, and λ is a scaling weight. In all experiments, $\lambda = 1$.

Thus, our solution combines ICM for reward learning and reinforcement learning. The full CDIL training procedure is summarized in Algorithm 2.

Algorithm 2 Curiosity-driven imitation learning (CDIL)

1: **Input:** Expert demonstration data $\mathcal{D} = \{(s_i, a_i)\}_{i=1}^{N}$.
2: Initialize policy π, *encoder* q_ϕ and *decoder* p_θ.
3: **for** $e = 1, \cdots, E$ **do**
4: Sample a batch of demonstration $\tilde{\mathcal{D}} \sim \mathcal{D}$.
5: Train f_{θ_F} and g_{θ_I} to optimize the objective (5) on $\tilde{\mathcal{D}}$.
6: **end for**
7: **for** $i = 1, \cdots,$ MAXITER **do**
8: Update policy parameters via any policy gradient method, e.g., PPO on the intrinsic reward inferred by Eq. (6).
9: **end for**
10: **Output:** Policy π.

In brief, the process begins by training ICM for E epochs (Steps 3-6). In each training epoch, we sample a mini-batch of demonstration data $\tilde{D}$ with a size of B and maximize the objective in Eq. (5). Steps 7-9 perform policy gradient steps, e.g., PPO(Schulman et al., 2017), so as to optimize the policy π with the intrinsic reward r_t inferred with ICM using Eq. (6). We treated CDIL as a related baseline in our experiments, using the feature extractor with the same architecture as the *encoder* except for the final dense layer. We trained the ICM using the Adam optimizer (Kingma & Ba, 2015) with a learning rate of 3e-5 and a mini-batch size of 32 for 50, 000 epochs. In each training epoch, we sample a mini-batch data every four states for Atari games and every 20 states for continuous control tasks.